INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO
MESTRADO EM ENGENHARIA INFORMÁTICA




Aplicação de métodos ágeis fora da indústria de software

BRUNO ALEXANDRE CARDOSO GUEDES
Outubro de 2018

POLITÉCNICO
DO PORTO

APPLYING AGILE OUTSIDE OF THE SOFTWARE INDUSTRY

Bruno Guedes

Porto 2018

APPLYING AGILE OUTSIDE OF THE SOFTWARE INDUSTRY

Bruno Guedes

Dissertation
at Instituto Superior de Engenharia do Porto
Graphic and Multimedia Systems - Master's Degree
Porto

Submitted by
Bruno Guedes
from Porto

Porto, 09/10/2018

First Advisor: Dr. Nuno Escudeiro
Second Advisor: Dr. Mafalda Ferreira
Porto, 09/10/2018

Index

Abstract **xiii**

1 Introduction **1**
1.1 Context . . . 1
1.2 About the problem . . . 1
1.3 Motivation . . . 2
1.4 Goals and Results . . . 2
1.5 Added Value . . . 3

2 Problem Context **5**
2.1 The Problem . . . 5
2.1.1 Business Concepts . . . 5
2.1.2 Barriers . . . 6
2.1.3 Analyzing Real Scenarios . . . 8
2.2 Problem Definition . . . 13
2.3 Customer Segments . . . 14

3 State of the Art **25**
3.1 Agile . . . 25
3.2 SCRUM . . . 26
3.2.1 SCRUM Pillars . . . 26
3.2.2 SCRUM Roles and Concepts . . . 27
3.2.3 SCRUM Formal Events . . . 27
3.3 SCRUM Nexus . . . 28
3.3.1 Where and why does SCRUM not fit? . . . 29
3.3.2 What is SCRUM Nexus? . . . 29
3.4 Kanban . . . 32
3.4.1 The Kanban Values . . . 32
3.4.2 The Kanban Board . . . 35
3.4.3 Little's Law . . . 36
3.4.4 Kanban General Practices . . . 36
3.4.5 Kanban Roles . . . 38
3.5 Current Agile Technologies . . . 38

3.5.1 Jira . . . 38
3.5.2 Trello . . . 39
3.5.3 Asana . . . 39

4 Value Analysis **41**
4.1 NCD . . . 41
4.1.1 The Engine . . . 42
4.1.2 Five Elements of the Activity . . . 42
4.2 Value For The Costumer . . . 44
4.3 Value Proposition . . . 45
4.4 Canvas Model . . . 45
4.5 AHP . . . 46

5 Sprint Task Methods **51**
5.1 Methodology Concepts . . . 51
5.2 A New Agile Concept . . . 53
5.3 Methodology Flow . . . 59
5.4 Platform Approach . . . 61

6 Solution **63**
6.1 Use Cases Overview . . . 63
6.1.1 Global Overview . . . 63
6.1.2 Time . . . 65
6.1.3 Managers . . . 66
6.1.4 Collaborators . . . 66
6.2 Sprint Task Platform . . . 67
6.2.1 Platform Architecture . . . 67
6.2.2 Authentication System . . . 69
6.3 Deploying Sprint Task . . . 72
6.4 Approaching Mobile Users . . . 74
6.5 Continuous Delivery System . . . 74
6.6 Business Continuity Plan (Informal) . . . 75
6.6.1 Beta Phase . . . 75
6.6.2 Production Phase . . . 76
6.7 Positioning Sprint Task With GDPR . . . 77

7 Evaluation **79**
7.1 Testing Sprint Task Methods . . . 79
7.2 Hypotheses . . . 87

8 Conclusions **89**

Acknowledgements **93**

Attachments 95

List of Figures

2.1 Fabsgarage 8
2.2 FAP 9
2.3 ALopes Industry 11
2.4 ALopes Studio 12
2.5 The Problems 13
2.6 Developer 15
2.7 Development Team 16
2.8 Product Manager 17
2.9 Collaborator 18
2.10 Quality Control Team 19
2.11 Repair Technician 20
2.12 Technicians team 21
2.13 Manager 22
2.14 Company Board 23

3.1 SCRUM formal events 28
3.2 SCRUM Nexus work flow 30
3.3 Kanban Values [1] 33
3.4 Kanban Board [1] 35
3.5 Little's Law[1] 36
3.6 Kanban General Practices [1] 36

4.1 NCD [2] 41
4.2 Canvas Model 45
4.3 Which criteria is more important? [6] 47
4.4 Client Orientation Alternatives Evaluation [6] 47
4.5 Adaptability Alternatives Evaluation [6] 48
4.6 Standardization Alternatives Evaluation [6] 49
4.7 AHP Result [6] 49

5.1 Sprint Task methods solutions 51
5.2 Projects and Tasks Relationship 53
5.3 Active Sprint Concept 55

5.4 Retrospective Concept . . . 56
5.5 Application Concept . . . 57
5.6 Form Concept . . . 58
5.7 Process Diagram . . . 59
5.8 Project Structure . . . 61
5.9 Task Structure . . . 62

6.1 Use Case Diagram . . . 64
6.2 Renew Sprints Procedure . . . 65
6.3 Managers Core Responsibilities . . . 66
6.4 Collaborators Core Responsibilities . . . 67
6.5 Platform Architecture . . . 68
6.6 User Validation Process . . . 70
6.7 Deployment Diagram . . . 73
6.8 Continuous Delivery . . . 74
6.9 Business Continuity Plan - Informal Scheme . . . 76

7.1 Procedure Efficiency Form . . . 80
7.2 Collaborator Efficiency Form . . . 81
7.3 Manager Planning Efficiency Form . . . 82
7.4 Measuring Plan Efficiency . . . 84
7.5 Measuring Plan Efficiency . . . 85
7.6 Measuring Procedures Efficiency . . . 86

8.1 Projects page . . . 95
8.2 Project quick view . . . 96
8.3 Project sprint . . . 97
8.4 Task overview . . . 98
8.5 Project attachments . . . 99
8.6 Teams page . . . 100
8.7 Team overview . . . 101
8.8 Collaborator overview . . . 102
8.9 Sprint planning - 1 . . . 103
8.10 Sprint planning - 1 . . . 104
8.11 Sprint planning - 2 . . . 105
8.12 Sprint planning - 3 . . . 106
8.13 Plus button overview . . . 107
8.14 Create project form . . . 108
8.15 Project Sprint . . . 109
8.16 Project attachments . . . 110
8.17 Teams overview . . . 111
8.18 Team view . . . 112
8.19 Team sprint . . . 113

8.20 Sprint view 114
8.21 App center view 115
8.22 Create project 116
8.23 Create sprint type 117
8.24 Chang sprint type state 118
8.25 Create team 119
8.26 Plan sprint 120
8.27 View active sprint 121
8.28 Move task 122
8.29 Data updater structure 123
8.30 Update manager colleagues 124
8.31 Update teams 125
8.32 Update sprint types 126
8.33 Update artifacts 127
8.34 Form overview 128

List of Tables

2.1 Business Concepts - part 1 5
2.2 Business Concepts - part 2 6

3.1 SCRUM roles and concepts 27
3.2 Nexus Process Flow 31
3.3 Kanban Values 34
3.4 Kanban Genral Practices 37

4.1 AHP Alternatives 46

5.1 Concepts Contextualization - part 1 52
5.2 Concepts Contextualization - part 2 53

Abstract

This dissertation explores the lack of organization experienced in medium and small companies work-flows, and aims to build an agile methodology directed to non-software organizations. Adopting an objective perspective, a study conducted in a group of companies, extracts flaws to suggest solutions that solve inefficient practices. In addiction, this dissertation uses the referenced study as a base to create Sprint Task, an agile methodology and platform for non-software environments.

Fixing the lack of constant organization in non software companies and creating an agile environment on their projects, were the main reasons that led to the development of Sprint Task concept. The development process is described through this dissertation.

Also, the present document purposes to prove that general common industry work-flow environments are improved by Sprint Task methods and that the Sprint Task platform is versatile and adaptable to different business areas.

Introduction

1.1 Context

Software development methodologies have suffered significant changes since the introduction of the Agile Manifesto. These changes have affected software development strategies and developers perspective regarding how software is developed.

Before the Agile Manifesto, software was projected as a single project-plan concept, but now, the software industry is focused in a client-oriented perspective. As a consequence, this approach has benefited each stakeholder in the development process. Specifying, the Agile Manifesto brought cost's reduction, work motivation, organization, documentation standards and client satisfaction.

Besides the positive impacts of the Agile Manifesto in the software industry, its direct impact on the general industry was not relevant.

1.2 About the problem

The concept of productivity is always related to organization and preparation. In a work-flow environment, non-standard and non-prepared procedures are usually slower, unsustainable or unfeasible.

As a consequence of the Agile Manifesto, the software industry has suffered significant positive changes. The introduction of agile methodologies, such as SCRUM and Kanban, caused a big impact on the industry and it changed the way software was planned, developed and maintained.

From the perspective of a developer, the project dynamics awareness is increased with agile development methods, since the developer knows what to do, what his team is doing and where to get or ask for help. This results from an objective and practical communication, along with the increased responsibility and autonomy caused by agile methodologies.

However, collaborators are not always aware of the project dynamics on medium and small companies of non-software industry. In fact, most of them have an impulsive, hour-based and hierarchic dependent mandatory work-style. Usually, a work environment with these characteristics have collaborators with low responsibility, freedom restriction and strict procedure guidelines, which consequently causes a low efficiency rate and a company growth stagnation.

Aiming to increase productivity, companies create their own organization and procedure-flow standards, due to the uniqueness of their usual tasks. Working efficiently is a chain process that can always be optimized and restructured, to improve the product quality.

1.3 Motivation

As a service provider company, ALopes Industry is focused on the technical assistance of quality control systems, on the textile industry. With the recent company growth, the work-flow methods became disorganized and unsustainable for the current company structure and planning practices. The amount of non-urgent pending service requests and the output capacity gap, was increasing every day. Customers were requesting more services and ALopes Industry work-flow methods needed to be reinvented.

1.4 Goals and Results

The present dissertation aims to present the project development with an objective and modular approach. Each phase of the project is presented independently, with the following goal order.

- Analyze the work-flow practices of a group of companies
- Develop an agile methodology, focused on non-software companies
- Develop a software platform to support the created agile methods
- Plan a testing approach

At the end of the project, the artifacts expected are a stable documented prototype of the agile methods, a software prototype of the platform that will support the methodology, and a defined group of tests, based on the work-flow analysis results.

1.5 Added Value

The impacts of agile methods affects companies practices and collaborators. An organized work environment is more stable and predictable. Collaborators are motivated to be autonomous and to work in a collaborative and supportive way, creating a biological work-flow that has the ability to react and overcoming unpredictable situations, on its own.

Problem Context

2.1 The Problem

2.1.1 Business Concepts

The present dissertation aims to document the development of the Sprint Task agile concept and platform. Aiming to introduce the agile methods into several areas of the common industry. As a complex project, it is important to explain key business concepts and specify an unique definition for each one. The following table is responsible for clarifying the meaning of each concept when related to the Sprint Task project.

Common Industry	In the context of this project, non-software industries that work in a project-driven perspective, are known as common industries. A few examples of common industries are car repair shops, on site machine repair companies and furniture production lines.
Manager	A manager is a role of leadership that is required in a structured work-flow. This role is responsible for assigning task's, creating projects, planning Sprints, etc...
Collaborator	Collaborators are managed by the Manager and are usually specialized in specific parts of the production line or work-flow.
Team	A team is a group of collaborators that is responsible for a specific area of the work-flow. Teams are also managed by the Manager.
Work Process	The work process is a chain of events, collaborators, teams, managers and clients that are part of organizations daily behaviour.

Table 2.1: Business Concepts - part 1

Discipline	This concept is based in the daily work-flow that naturally "creates well-organized memories, history, and experience" [5].
Agility	Related to the discipline concept, agility describes the application of memories "and history to adjust to new environments, react and adapt, take advantage of unexpected opportunities, and update the experience base for the future" [5].
OIP	Optimizable Interest Point - A specific work process procedure that can be optimized.

Table 2.2: Business Concepts - part 2

2.1.2 Barriers

The introduction of Sprint Task into the market will meet multiple obstacles. In order to prevent and reduce their impact, those were analyzed so that they could be taken in consideration during the project development. This approach will ease the creation of timed planned approaches to introduce Sprint Task into the market.

Skepticism

Skepticism is an enemy of innovation and is felt in the software industry too. As quoted in 2002 by Pekka Abrahamsson, Outi Salo, Jussi Ronkainen and Juhani Warsta, "industrial software developers have become skeptical about new solutions that are difficult to grasp and thus remain not used".

Being skeptical to change is usually related to old and stagnated companies. This being a characteristic of the most medium and small companies in Portugal, it represents one important barrier in the project development process.

To overcome this obstacle, it's required to make the costumer segments understand that working process errors on the production line will have future unwanted impacts on the product, the production team or in the company itself.

These companies need to understand the benefits of Sprint Task concept, to know how to use it and most important, how to benefit from it. In fact, it requires training to efficiently improve agile techniques and adapt them to each particular context. But once they are mastered, the work process will be improved with agility and the discipline will be more consisted and organized.

Usability Issues

The adaption phase to new methods is always difficult in a work environment. Different ways of thinking and different procedures can be hard to follow in an early phase of the concept introduction. In order to avoid this problem, Sprint Task platform needs to be adapted to a general user concept. It is important to reduce usability issues, and for that, the platform must be designed with the following aspects taken in consideration:

- Mindset divergences
- Cognitive usage barriers
- Cognitive awareness

Reality Application

Every work-flow is different because it belongs to an unique company environment. For that reason, applying Sprint Task methods will always require a work-flow analysis and study, in order to make the methods usage as efficient as possible.

Although Sprint Task guide is an important guideline to a successful implementation, it is not a standard solution. Each company has different needs and different processes that need to use unique tools and management styles. An organized and planned usage of the methods is crucial, otherwise the complexity added to the work-flow system will be useless and expensive.

Integrating an agile method into a work-flow, will always require an adaption phase for procedures to run smoothly. Every company environment is different and management solutions are not standard, it is not possible to create a mandatory guideline for Sprint Task methods.

Despite the Sprint Task guide, different environments require different approaches and tools. The versatility of the platform, provides an easy way to centralize all the software tools in the same system. Each company will have a group of dedicated applications, which can be developed for a specific company and integrated with Sprint Task main system. With this approach, Sprint Task allows companies to centralize and standardize documents, processes, communications and artifacts.

2.1.3 Analyzing Real Scenarios

Before creating Sprint Task methods, the work-flow of several organizations was observed and studied, in order to uncover the most common problems of medium and small companies work-flows. Due to the different companies areas included in this study, each description will be simple, informal, objective and following the same structure.

- Company presentation
- Usual process
- Collaborators complaints
- Work-flow flaws

Fabsgarage

Fig. 2.1: Fabsgarage

The first observed company is a small car repair shop, focused in providing auto repair and maintenance services to a large variety of costumers. Fabsgarage work-flow is usually dependent of the manager pre-planning procedures and, although there is just one person responsible for the planning tasks, its impact in the work process is high.

Usually, Fabsgarage clients reach the company when they have a technical problem or a maintenance appointment with their car or motorcycle. After the first contact, the technician assigned to the service realizes a pre-prepared standard checkup procedure to the car. After this procedure, the results are communicated to the costumer with a modular budget and the car checkup documentation is archived in the physic vehicle folder. In other words, Fabsgarage is adopting a vehicle oriented, handwritten organization that allows the company to objectively store the vehicle history.

It became clear, during several meetings with the manager and collaborators, that the most important work-flow complaints from the company were about the checkup procedure. It was unanimous that the checkup procedure documentation is too time consuming and it

is not uncommon to loose the checkup sheet over time. However, this particular procedure is a useful source of information for the company and a straight-forward way to present the vehicle state to the costumer, the usage of unnecessary resources is always a barrier to every work-flow and an OIP.

Besides the company perspective, once comparing Fabsgarage work-flow observation with the general extracted results, it became clear that their flaws are not different from the average observed problems.

In Fabsgarage, the information available is limited to the checkup procedure and the human search capacities. Every time a technician needs to access a vehicle history, he needs to physically search the respective archive and find the respective checkup sheet. Although the documentation is standardized, it is only possible to access the checkup procedure and it usually takes a considerable amount time per search. Moreover, documenting part of the process may not be enough for all situations. Agility moments are undervalued, unreproduced and cause disorganization over the collected information. For example, it is usual for technicians to take pictures of specific parts, so that it is easier to reproduce certain procedures or document the current state of the vehicle. However, this information is not stored anywhere and is frequently lost as soon as the service ends.

FAP

Fig. 2.2: FAP

FAP is furniture production company that is focused in creating unique solutions for costumers, according to their requests. Althought FAP has more tha a decade of story, its management style did not evolve accordingly.

Collaborators have strong hierarchic dependencies, managers are not able to create accurate plans and usually uses impulsive approaches with its collaborators and clients.

Usually, FAP clients are ask for a budget of a custom furniture piece. Then, the management department procceds to evaluate the most efficient way to build the referred piece and its price.

If the client accepts the budget, the piece follows the usual manufacturing process which starts at the cutting section.

The cutting section is responsible for preparing each part of the piece and realizing quality tests over those.

Once every part of the piece is prepared, the polish section is responsible for coloring the piece according to the client details.

Finally, the montage department tests the piece for imperfections by mounting the piece one last time before delivering it to the costumer.

Altough FAP work-flow procedures look organized and well established, it has been failing in the last few years. Some of the main detected problems are caused by the lack of planning from the management department and the lack of discipline of collaborators.

The first main problem is caused by the lack of documentation of each furniture piece. There are no documents or information cappable of informing the different interventions and problem detections. Each step of the production line is triggered and accepted by a manager.

The second main problem is related to the excessive responsabilities aggregated by managers. Each manager gathers administrative functions, production roles, people management responsabilities and are also responsible for the client support.

The third and last main problem is caused by the lack of collaborators discipline. In general, they are reckless and not autonomous. Usually, a collaborator is not active until receiving direct orders from a manager and even when handling simple tasks, their lack of effort is frequently noticeable.

Besides the previously highlighted problems, there are also a few problems in the work process. For example, there is no quality evaluation or documentation of a furniture piece upon a department transaction.

ALopes Industry

Fig. 2.3: ALopes Industry

Applying its efforts in the textile industry, ALopes Industry is focused in providing technical support and maintenance services on quality control automated machines, included in production lines and laboratories. Representing Mahlo and Mathis on the Iberian Peninsula, ALopes Industry usually practices on site repair actions, due to the importance of quality control systems on textile production lines.

Regarding its size, ALopes Industry has a small valid structure that has been growing over the years. At the moment, it counts with three collaborators and one of them is also the manager.

Clients reach ALopes Industry by contacting either the manager or one of the collaborators, by communicating a problem with a specific machine or asking for technical guidance on the production process. A pending visit is created and added to the pending tasks archive. By doing this, ALopes Industry maintains an updated backlog that eases the planning process. In order to maintain a sustainable intervention plan over time, and once the majority of the interventions are on the clients facilities, ALopes Industry usually groups interventions by priority and region, allowing technicians to solve the most priority cases and maintaining their work trips efficient.

As stated by the company members during the observation period, current software solutions are not easy to use and do not fit in ALopes Industry market approach. Their flow is not versatile, the provided tools are difficult to use and the software interface is confusing and distracting. Due to these reasons, ALopes Industry maintains physical organization archives for each machine with the respective documentation, visit sheets and agility moments.

Despite the existent correct organization in the company documentation, there were some barriers experienced during the observation.

Firstly, the information is not accessible from everywhere and, without the access to previous interventions documents, certain procedures may take significantly more time to complete.

Secondly, there were different software tools in the company work process that contributed for the information dispersion and disorganization. Technicians were searching for specific data on the wrong places and were confused about how and where to document

certain procedures.

Thirdly, there are no documentation standards defined and collaborators usually create their documentation freely. This makes the search process hard and collaborators unaware of the documentation existence. Lastly, planning procedures are usually done in a daily basis due to the frequency of urgent interventions. There is no weekly plan or objective defined which does not help pursuing medium-term goals over time.

ALopes Studio

Fig. 2.4: ALopes Studio

Born from ALopes Industry, ALopes Studio is a software company that is focused in building customized solutions for clients and create software products for distribution. However having a small structure, a designer and a developer, ALopes Studio handles multiple projects simultaneously, including the development of Sprint Task.

ALopes Studio is a company dedicated to software and design, and although it is not the focus of the Sprint Task itself, it was integrated in the observation to analyze work-flow flaws in a small software team, when handling multiple projects at once.

Adopting Kanban as the active agile methodology in the company, ALopes Studio has been able to include multiple projects into their sprints. However, there are a few downsides to the Kanban philosophy when adopting multiple projects.

Including several projects into a Kanban board demands a visual separation between each project in order to allow a visual comparison between each one. However this separation provides a way to compare project states, it is not trustworthy to measure the project development over time with a visual comparison.

The more complex projects are, more complex the Kanban board is and more populated it will be. Managing a full Kanban board with different dependencies, development paces and deadlines, is not an easy task for the manager that holds the planning responsibility.

The difficulty of managing different project backlogs on a full Kanban board is high and increases when managers are required to manage priorities with limited resources. This is a problem experienced in ALopes Studio during the development of print Task, as

it became hard to predict deadlines and due dates due to the non stable tasks priority of other projects.

However, the hardest part about using Kanaban in ALopes Studio environment, was the non quantitative and non qualitative overview of each project. At the end of each sprint, it was hard to measure how much the project had grew in that sprint. As different source tasks priority changed over time, so did the sprint goals. Consequently, if sprint goals changed during that specific sprint, its success rate evaluation was compromised due to the environment instability. It was not possible to determine if a project-sprint plan was successful or not because only the sprint overview was trustworthy.

2.2 Problem Definition

Although the problem has been already defined, it's important to analyze how the standard problems in an objective and standard way. To do this, the main problems were schematized in the next figure and will be explained and justified upfront.

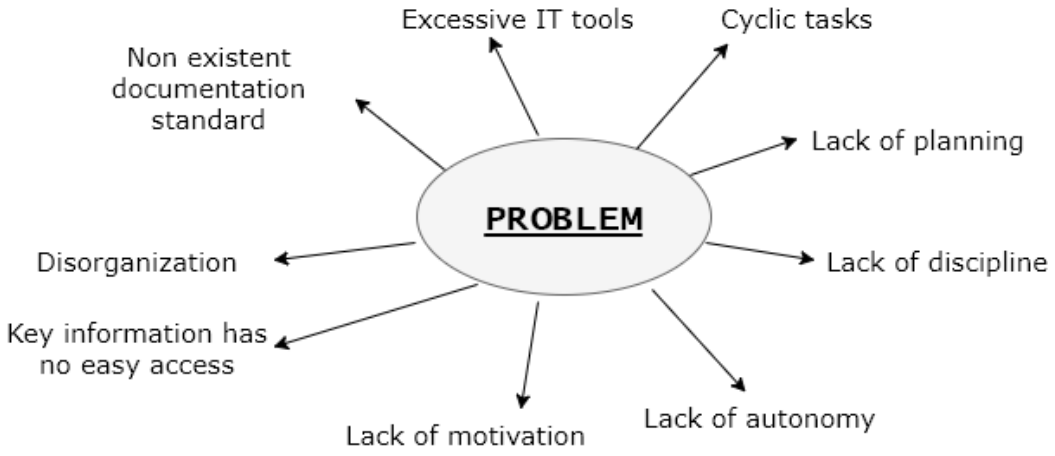

Fig. 2.5: The Problems

Apart from the visual perspective, it is important to clarify the identified problems.

- **- Excessive IT Tools:** It's common that in most of the companies, the IT tools used in support of their activity are decentralized and not connected. This is a problem that does not help efficiency and the standardization of the information.
- **- Cyclic Tasks:** In most of the production lines of medium and small companies, the task's to realize are cyclic and repetitive. In a long-term point of view, collaborators effort and focus on the task tend do decrease. This problem is an important aspect in a work process and is a point of interesting of the concept.

- **- Lack of Planning:** Old businesses work as they are used to. The planning method is usually simple and unclear. It's usually done in a spontaneous way and dependent of an hierarchic order.
- **- Lack of Discipline:** This discipline is not the same as the referred on the business concepts. The lack of discipline pointed in the scheme is related to the responsibility sense of the work process stakeholders. It was observed that, not in an intentionally way but, they usually don't have the initiative to do certain easy and quick task's which can influence future decisions in the production. Quick examples of this are missing appointments due to the lack of agenda usage.
- **- Lack of Autonomy:** In the analyzed companies, collaborators experience lack of autonomy when deciding what to do. They usually wait for an hierarchic order to start working, which is essential in some cases but, it's lack of productivity in most of them.
- **- Lack of Motivation:** Accommodated collaborators are common in old businesses. This is a barrier to the implementation of new systems and also a good starting point to stagnation. Motivation should be implicit in the work flow and process. The need of motivation factors is also a point of interest of this project.
- **- Key Information Without Easy Access:** Using the discipline concept, when a specific information is needed for some reason, collaborators feel great difficulties to find it. This happens because of the decentralized information store habits or because only physical information exists.
- **- Disorganization:** In a work process, each person or event is a chain link. Chain links should be linked and not separated from each other. Information should be clear to any link and this is not something easy to happen in the common industry. Normally, each collaborator is a standalone link which is focused in his task.
- **- Non Existent Documentation Standard:** Reusing discipline and agility concepts in a common industry context, it became clear during the companies analysis that, whenever there was an agility event capable of improving the team's discipline, the information didn't flew. Even if the event was documented, it would be in a non-standard form and eventually lost and forgotten.

2.3 Customer Segments

During the initial phase of this project, and since the costumer segments was defined from the beginning, the more common stakeholders of the work process were turned into personas and analyzed in an objective way.

Developer

Developers will benefit from the union of multiple projects in the same platform and in the possibility of a personal sprint planning.

Besides of the methodological organization needs of the development process, documenting task will be a new added value because the task documentation will updating the current project documents and ready to compare to the old project documentation.

Fig. 2.6: Developer

Benefits:

- Get to know the state of every project the developer is part of
- Plan is own work flow during the Sprint
- Share updates during the Sprint
- Consult documentation from previous tasks
- Associate documentation to tasks

Development-Team

Development teams will be working on a concept of self sufficiency and methodological logistics. Teams will have more and better data to communicate, decide and intervening without the Project Manager.

Fig. 2.7: Development Team

Benefits:

- Being updated of the current Sprint state
- Predict difficulties in specific tasks to mobilize resources earlier
- Review tasks in a simple and objective way
- Simpler retrospective events
- Make code-review, quality control and test's processes simpler

Product Manager

The added value for the product manager resides in the increased amount and quality of the information about each project task. This information will help in the product manager control process and will give more sustainability to his analysis and decisions over projects and work process.

Fig. 2.8: Product Manager

Benefits:

- Being up to date about the state of multiple projects and multiple Sprints
- Analyze teams productivity and gather data about their improvement
- Increase predictability
- Sustainable and important interventions are simplified
- Accurately delegate tasks

Collaborator

Collaborators will see their autonomy being more important over time and the reduction of the hierarchic dependencies. In many cases, collaborators are not aware of the importance of their contribute even being crucial for organizations for their knowledge on the production process. Collaborators are the pillars of an organization.

Fig. 2.9: Collaborator

Benefits:

- Get to know their Sprint tasks
- Reduced hierarchic dependencies
- Defined goals
- Being self-aware of their impact in the production
- Having team oriented goals

Quality Control Team

The quality control will benefit for the standardization and quality of the information gathered during the Sprint's. It will be easier to verify procedures analyze documentation and evaluate the preformed work.

Fig. 2.10: Quality Control Team

Benefits:

- Know what to analyze
- Available detailed procedures information
- Standard results communication

Technician

For technicians, the added value resides in the easy and fast access to information. This is important due to the simple organization and utilization of his tools and documentation influence in their tasks.

Fig. 2.11: Repair Technician

Benefits:

- Pre-prepare tasks
- Easy access to appropriate documentation
- Access to clients history
- Access to machines history
- Fast, easy and standard repairs documentation

Technicians Team

Teams of technicians can be motivated by being aware of the achieved goals importance. This will promote the improvement of self-sufficiency and pro-activism inside of teams.

Fig. 2.12: Technicians team

Benefits:

- Overtime analysis of team performance
- Qualitative analysis of the used procedures on a mid and long-term perspective
- Self-motivation
- Creation of learning mechanisms for new collaborators or old collaborators from different areas
- Fast, easy and standard repairs documentation

Manager

Team managers, with a structured and adapted use of the concept, will benefit from quantitative and qualitative data about their teams to support sustainable decisions and approaches. To standardize and automate procedures, the platform will work as a tool to benefit the work process stakeholders and, as consequence, the team manager.

Fig. 2.13: Manager

Benefits:

- Real-time, quantitative and qualitative team performance analyses
- Analogous team performance comparison
- Implementation of new approaches and procedures based in reliable work-process data
- Control of teams performance with Sprint Reviews, Sprint Retrospectives and Sprint Planning
- Create and plan Sprints
- Assign tasks to teams
- Improve planning skills
- Improve product quality

Company Board

The platform's data management allows the company board to easily consult and analyze technical data which will contribute to the companies overview in an objective way. With the use of the work process data, companies can analyze how their objectives are being accomplished and improve less efficient areas.

Fig. 2.14: Company Board

Benefits:

- Know how achievements are being accomplished
- Extraction of technical information, archiving procedures and easy information access
- Assisted budget and investment planning
- Simplify the conjugation of commercial, technical and client satisfaction goals

State of the Art

3.1 Agile

A stable and consistent work process and problem approach is essential for the success of every organization. The work process stability and consistency is a consequence of discipline [5]. A work process is more stable and consistency as more organized are memories, history and experience.

However, a work process is never perfect and it can be improved by agility [5], the application of memories, history and experience to take advantage of unexpected opportunities. Discipline [5] is improved agility moments and, as consequence, so does the work process.

According to Highsmith and Cockburn, "what is new about agile methods is not the practices they use, but their recognition of people as the primary drivers of the project success, coupled with an intense focus on effectiveness and maneuverability" [10] [7].

After the agile manifesto, the work process of the software industry started to value people over tools, effectiveness over usual work flow practices and products over contracts. The following list contains the agile manifest values [3].

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

Agile methods are a benefit to the work process even though they are not easily adapted to every organization, particularly in non-software companies.

3.2 SCRUM

As stated in SCRUM Guide, "SCRUM is a framework for developing, delivering, and sustaining complex products" and is defined as "a framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value" [12].

Being used since the 1990's, SCRUM is an agile methodology meant for software development teams to continuously improve their work process and their product with "roles, events and artifacts, governing the relationships and interaction between them". [12]

3.2.1 SCRUM Pillars

SCRUM is based in an iterative and an incremental approach to optimize predictability and control risk and is supported by three pillars, which are transparency, inspection and adaption.

- **- Transparency** requires that significant aspects of the process must be visible to those responsible for the outcome to be defined by a common standard [12].
- **- Inspection** SCRUM artifacts and progress towards a Sprint Goal will help teams to detect undesirable variances [12].
- **- Adaptation** is a consequence of inspection and undesirable variances. It's the possibility to adapt resources in order to reduce the impact of unpredictable situations [12].

3.2.2 SCRUM Roles and Concepts

It is important for everyone who is not familiar with SCRUM to understand these roles and concepts. All the concepts and roles are described in the table bellow.

Concept	Definition
Product Owner	"The Product Owner is responsible for maximizing the value of the product resulting from work of the Development Team. How this is done may vary widely across organizations, Scrum Teams, and individuals." [12]
Sprint	"The heart of Scrum is a Sprint, a time-box of one month or less during which a "Done", useable, and potentially releasable product Increment is created. Sprints have consistent durations throughout a development effort. A new Sprint starts immediately after the conclusion of the previous Sprint." [12]
Scrum Master	"The Scrum Master is responsible for promoting and supporting Scrum as defined in the Scrum Guide. Scrum Masters do this by helping everyone understand Scrum theory, practices, rules, and values." [12]
Scrum Team	"The Scrum Team consists of a Product Owner, the Development Team, and a Scrum Master. Scrum Teams are self-organizing and cross-functional. Self-organizing teams choose how best to accomplish their work, rather than being directed by others outside the team. Cross-functional teams have all competencies needed to accomplish the work without depending on others not part of the team." [12]
Product Backlog	"The Product Backlog is an ordered list of everything that is known to be needed in the product." [12]
Sprint Backlog	"The Sprint Backlog is the set of Product Backlog items selected for the Sprint." [12]

Table 3.1: SCRUM roles and concepts

3.2.3 SCRUM Formal Events

Regarding the formal events, they intend to formalize a cyclic process in a Sprint and they are meant to be objective, efficient and helpful. They require responsibility and cooperation by all the involved parts and are intended to return important artifacts, which are required to the sustainability of the SCRUM process and the product healthy development.

The **Sprint Planning** is about planning a Sprint and it's done with the entire SCRUM Team and is meant to define the deliverables for the next iteration [12]

.

The **Daily SCRUM** meeting is a 15-minute time-boxed meeting for the development team and his held everyday of the Sprint. It's at this meeting that the development team plans the work for the next 24 hours [12]

.

The **Sprint Review** is held at the end of the Sprint to inspect the increment and adapt the Product Backlog if needed. It's also at this event that it's discussed what are the next things to do, in order to increase the product value [12]

.

The **Sprint Retrospective** it's an opportunity to inspect the Scrum Team with the objective of improving the downsides of a Sprint. This event is done after the Sprint Review and it's the opportunity to prevent undesirable events in the next Sprints [12]

.

In the image bellow we can see how SCRUM formal events are connected and their order.

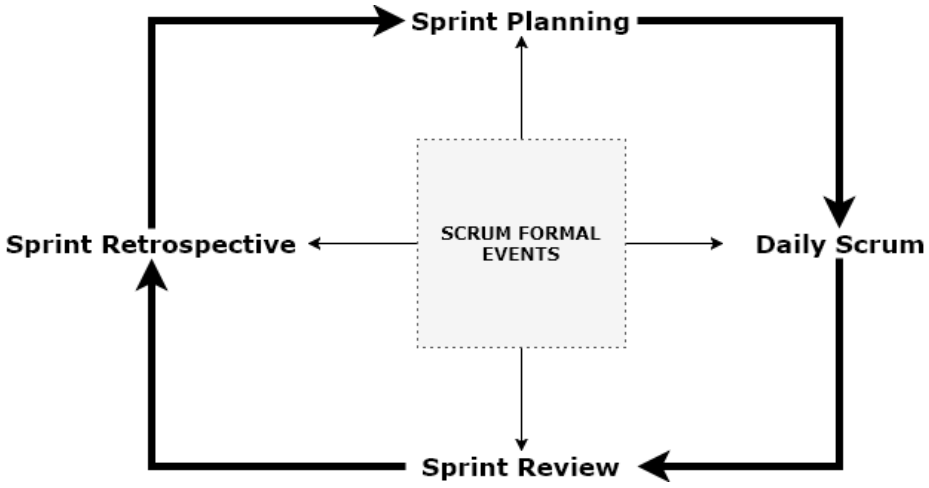

Fig. 3.1: SCRUM formal events

3.3 SCRUM Nexus

SCRUM is an agile methodology designed to manage SCRUM Teams collaboration. However, bigger projects are normally composed by multiple teams. This is a problem that SCRUM itself does not solve because, bigger teams means extensive meetings, inefficient conversations which usually results in disorganization and unmotivated developers.

To solve this, SCRUM Nexus uses "SCRUM as it's building block" [11] and it proposes to solve the problem of multiple teams working in the same Product Backlog in an efficient, objective and reliable way.

3.3.1 Where and why does SCRUM not fit?

The SCRUM Nexus guide refers that SCRUM can be an obstacle in projects which "more than one SCRUM Team is working off same Product Backlog and in the same codebase" [11]. Alongside the explanation, it also points to the difficulties with two important questions.
- "**If the developers are not on the same collocated team**, how will they communicate when they are doing work that will affect each other?" [11]
- "**If they work on different teams**, how will they integrate their work and test the Integrated Increment?" [11]
This problem clearly demonstrates that a good methodology is not enough to guide a project into success. It is required to be well applied and a compatible environment to be implemented in.

3.3.2 What is SCRUM Nexus?

"Nexus is a process framework for multiple Scrum Teams working together to create an Integrated Increment. Nexus is consistent with Scrum and its parts will be familiar to those who have used Scrum. The difference is that more attention is paid to dependencies and interoperation between Scrum Teams" [11] in every formal event.
An overview of the SCRUM Nexus workflow is available in the figure bellow.

It's implied in the previous figure, all the main changes to the SCRUM methodology. As enumerated in the Nexus Guide, we can enumerate analyze the changes in three perspectives.

- **Roles**: "A new role, the Nexus Integration Team, exists to coordinate, coach, and supervise the application of Nexus and the operation of Scrum so the best outcomes are derived. The Nexus Integration Team consists of the Product Owner, a Scrum Master, and Nexus Integration Team Members".
- **Artifacts**: "All Scrum Teams use the same, single Product Backlog. As the Product Backlog items are refined and made ready, indicators of which team will do the work inside a Sprint are made transparent. A new artifact, the Nexus Sprint Backlog, exists to assist with transparency during the Sprint. All Scrum Teams maintain their individual Sprint Backlogs" [11].

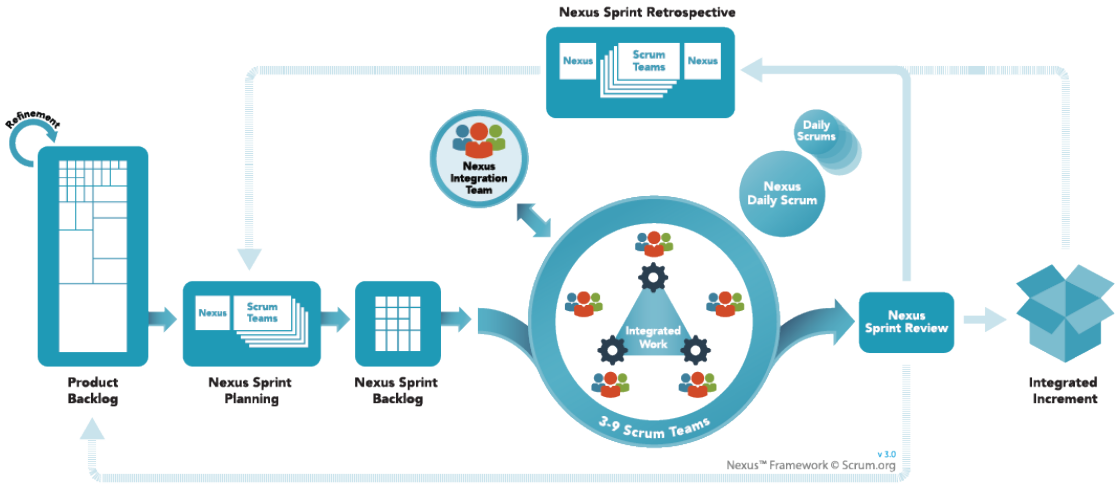

Fig. 3.2: SCRUM Nexus work flow

- **Events**: "Events are appended to, placed around, or replace (in the case of the Sprint Review) regular Scrum events to augment them. As modified, they serve both the overall effort of all Scrum Teams in the Nexus, and each individual team" [11].

As expected, SCRUM Nexus changes, comparing to SCRUM, have also a significant impact in the process flow. Described in the Nexus Guide, the changes were collected and organized in the table bellow.

Refine the Product Backlog	"The Product Backlog needs to be decomposed so that dependencies are identified and removed or minimized. Product Backlog items are refined into thinly sliced pieces of functionality and the team likely to do the work should be identified." [12]
Nexus Sprint Planning	"Appropriate representatives from each Scrum Team meet to discuss and review the refined Product Backlog. They select Product Backlog items for each team. Each Scrum Team then plans its own Sprint, interacting with other teams as appropriate. The outcome is a set of Sprint Goals that align with the overarching Nexus Sprint Goal, each Scrum Team's Sprint Backlog and a single Nexus Sprint Backlog. The Nexus Sprint Backlog makes the work of all Scrum Team's selected Product Backlog items and any dependencies transparent." [12]
Development Work	"All teams frequently integrate their work into a common environment that can be tested to ensure that the integration is done." [12]
Nexus Daily Scrum	"Appropriate representatives from each Development Team meet daily to identify if any integration issues exist. If identified, this information is transferred back to each Scrum Teams Daily Scrum. Scrum Teams then use their Daily Scrum to create a plan for the day, being sure to address the integration issues raised during the Nexus Daily Scrum." [12]
Nexus Sprint Review	"The Nexus Sprint Review is held at the end of the Sprint to provide feedback on the Integrated Increment that a Nexus has built over the Sprint. All individual Scrum Teams meet with stakeholders to review the Integrated Increment. Adjustments may be made to the Product Backlog." [12]
Nexus Sprint Retrospective	"Appropriate representatives from each Scrum Team meet to identify shared challenges. Then, each Scrum Team holds individual Sprint Retrospectives. Appropriate representatives from each team meet again to discuss any actions needed based on shared challenges to provide bottom-up intelligence." [12]

Table 3.2: Nexus Process Flow

The previous table enumerates an explains all the essential process flow changes implied by a Nexus guided project. The Scrum Team oriented perspective is visible in every mentioned concept and it allows the project to be decomposed and simplified, however with dependencies. Managing dependencies is an important and essential part of this methodology and it's responsibility of the Integration Team to "ensure the Scrum Teams within the Nexus understand and implement the practices and tools needed to detect dependencies"

and to deliver "increments of potentially releasable products" marked as "Done" [12].

3.4 Kanban

"Kanban is a method for defining, managing, and improving services that deliver knowledge work". It's described "as catalyst for rapid and focused change within organizations" in Kanban Condensed Guide [1].

Kanban "limits the amount of work in progress by using visual signals" to prevent work overflow and "ensure that the service works on the right amount of work" [1].

3.4.1 The Kanban Values

As well as SCRUM and SCRUM Nexus, Kanban is based in values which are pursued by the methodology process flow. According to the Essential Kanban Condensed Guide, the Kanban method "is motivated by the belief that respecting all of the individuals who contribute to a collaborative enterprise is necessary" [1].

If we observe the next image, we can notice that **Respect** is the last item of the list. This happens because **Respect** is the pillar of Kanban and it's implied in every single other value of the method.

This also implies that **respect, teamwork and collaboration** are keys for an efficient, sustainable and healthy project development.

The next image was taken from Essential Kanban Condensed and aims to represent Kanban values in a visual way.

Fig. 3.3: Kanban Values [1]

The following table aims to sum up the Essential Kanban Condensed guide values in an objective way and to define all the method values with their original definitions.

Transparency	"The belief that sharing information openly improves the flow of business value" [1].
Balance	"The understanding that different aspects, viewpoints, and capabilities all must be balanced for effectiveness" [1].
Collaboration	"The Kanban Method was formulated to improve the way people work together" [1].
Costumer Focus	"Knowing the goal for the system. Every Kanban system flows to a point of realizing value" [1].
Flow	"The realization that work is a flow of value, whether continuous or episodic" [1].
Leadership	"The ablity to inspire others to action through example, words and reflection" [1].
Understanding	"Primarily self-knowledge (both of the individual and of the organization) in order to move forward" [1].
Agreement	"The commitment to move together toward goals respecting - and where possible, accommodating - differences of opinion or aproach" [1].
Respect	"Valuing, understanding, and showing consideration for people" [1].

Table 3.3: Kanban Values

These are the values that Kanban Method was built on and they are the pillars of any Kanban system.

One of the most important and symbolic artifacts of a Kanban system is analyzed in the next topic, the Kanban Board.

3.4.2 The Kanban Board

The Kanban board is an essential tool and reference for the method. As described in Essential Kanban Condensed, "the board depicts a flow system in which work items flow through various stages of a process, ordered from left to right" [1].

The next image was taken from the referred Kanban Guide and it represents a typical Kanban board to be analyzed.

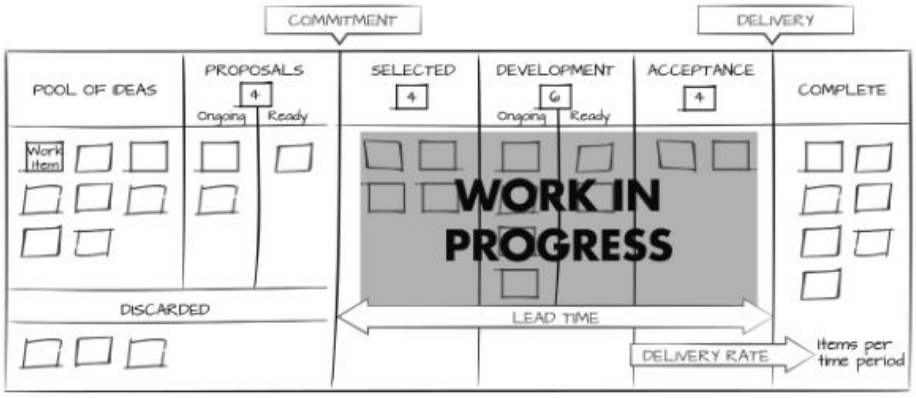

Fig. 3.4: Kanban Board [1]

"Kanban systems must have identified **commitment** and **delivery** points" in order to clearly define the work in progress state. This is important because of the limitations to impose to the work in progress phase that are usually visually signalized in the work in progress area [1].

These limitations exist to prevent work overflowing and a healthy and sustainable way to ensure the right amount of work.

There are some concepts that are important to have knowledge of when talking about Kanban.

Firstly, "the commitment is an explicit or tacit agreement with the customer and service" [1].

Secondly, it's essential to be aware about the definition of **delivery point**. This concept distinguishes accepted work from complete work. In other words, when a specific task is considered complete, it's ready to be delivered. This state change can be triggered by automated test's, work review or other procedures [1].

Lastly, **lead time** consists in the time frame between the commitment and delivery points [1].

3.4.3 Little's Law

"In a flow system that is not trending (and which all items that are selected are delivered) there is a simple relationship between the averages of these metrics over a specific period. It is known as Little's Law":

$$\overline{Delivery\ Rate} = \frac{\overline{WIP}}{\overline{Lead\ Time}}$$

Fig. 3.5: Little's Law[1]

This formula is essential for an accurate work in progress limitation.

Littles Law makes clear that, increasing WIP, without finding a way to increase the delivery rate, Lead Time will increase. However, decreasing WIP imply the reduction of Lead Time [9].

3.4.4 Kanban General Practices

As a methodology, there are General Practices of Kanban that are "essential for those managing Kanban systems". All the six general practices can be observed in the figure bellow [1].

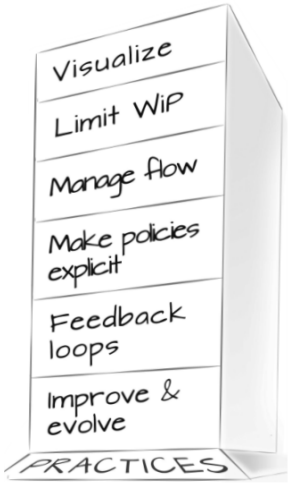

Fig. 3.6: Kanban General Practices [1]

All of this general practices involve "**seeing** the work and the policies that determine how it is processed", and "**improving** the process in an evolutionary fashion" [1].

The following table intends to explain and to make clear the definition of each General Practice of Kanban.

Visualize	"For it to be a Kanban systems rather than a simply a flow system, the commitment and delivery points must be defined, and WiP (Work in Progress) limits must be displayed to limit the work in progress at each stage between these points" [1].
Limit WiP	"Introducing and respecting limits on WiP changes a 'push' system into a 'pull' system, in which new items are not started until work is completed". "Observing, limiting, and then optimizing the amount of work in progress is essential to success with Kanban, as it results in improved lead time for services, improved quality, and a higher rate of deliverables" [1].
Manage Flow	"A key to understanding and maximizing the flow of value is the cost of delay of work items. This is the amount of an item's value that is lost by delaying its implementation by a specified period of time. In general, the cost of delay is a function of time, and the at which value changes, may or may not be constant over time."[1] Managing the Flow of a Kanban system is an essential task that can compromise the sustainability and healthy development of a project over time. A bad flow management can result in unsustainable projects.
Make Policies	"Explicit policies are a way of articulating and defining a process that goes beyond the work flow definition. A process expressed as work flow and policies creates constraints on action, is empowering within the constraints, and results in emergent characteristics that can be tuned by experiment." [1].
Feedback Loops	"Feedback loops are an essential part of any controlled process and are especially important for evolutionary change. Improving feedback in all areas of the process is important, but it is particularly so in **strategy alignment**, **operational coordination**, **risk management**, **service improvement**, **replenishment**, **flow** and **costumer services**" [1].
Improve and Evolve	"Kanban starts from the organization that it is now and uses the Lean flow paradigm (seeing work as a flow of value) to pursue continuous and incremental improvement" [1].

Table 3.4: Kanban Genral Practices

3.4.5 Kanban Roles

It is still important to clarify what are the roles which need to be assumed in a Kanban system and according to the Essential Kanban Condensed, they are only two.

"The **Service Request Manager** is responsible for understanding the needs and expectations of customers, and for facilitating selecting and ordering work items at the Replenishment Meeting."

"The **Service Delivery Manager** is responsible for the flow of work in delivering items to customers and for facilitating the Kanban Meeting and Delivery Planning".

3.5 Current Agile Technologies

Recent agile technologies that were designed to support agile methods have been gaining importance in the software industry work process. This document will only contains a light analyzes of Jira and Trello from Antlassian, and Asana.

3.5.1 Jira

Jira, owned by Antlassion, is an application focused on software development compatible with the use of SCRUM and Kanban. Jira is the most complete of the three analyzed tools and it requires a monthly team subscription.
As a project oriented tool, Jira has the possibility to generate artifacts and to be customized in the most efficient way for the costumer.

Having the strengths of being synchronized with Antlassian applications, such as Hip Chat and Bitbucket, Jira assumes and advantage position among Antlassian users.

Using the dashboard as an update and news center, Jira aims to centralize simple and useful information for their users and, once their users are mostly software developers and project managers, Jira uses a software related vocabulary and procedures.

Lastly, Jira connection to Bitbucket allows the developed code to be connected to the user stories. This is an advantage for software developers to detect and analyze code changes without extensive search or bureaucracy. Associated to this, software teams can create Software Versions and plan their development and Lead Time. Jira allows team members to check the progress of the versions and to connect them to Bitbucket.

3.5.2 Trello

Trello is a simple, efficient and organized board application with task related features that aims to improve team planning and team productivity over the Lead Time.

Being a single cross-platform project oriented adaptive board, with simple and fast usage allied with the usefulness of the objective created tasks, Trello assumes a strong position among developers.

Using an inner team perspective, Trello is a strong and reliable tool for Software teams to work with.

Also owned by Antlassian, Trello aims to satisfy a different costumer segment needs than with Jira. Focused on smaller projects, Trello has all the characteristics for spontaneous and fast fast agile planning.

3.5.3 Asana

Asana uses an out of the box approach. As a private company, Asana aims for "normal" users by providing tools to manage their small projects in an inner team perspective.

Asana do not force a methodology flow. In fact, Asana is based in a "can do" concept.

Once Asana provides tools and several ways to assign, share and plan task's, Asana does not force a work flow or requires actions to evolve.

Using a calendar view, Asana makes use of visual symbols to tell users about their projects.

Projects in Asana can be created easily and managed simply and effective. Their information will be joined into the dashboard and aims to alert the user about important facts regarding their associated tasks and projects.

Lastly, Asana provides a file management and conversations area. Aiming to use real-time communication and information sharing tools, Asana does focus in "normal" people and can, in fact, help non-software related users on their small projects.

Value Analysis

4.1 NCD

The new concept development model, known as NCD was created by Peter Koen and aims to help future decisions in an organization. This model will help the project to assume a sustainable and consistent position.

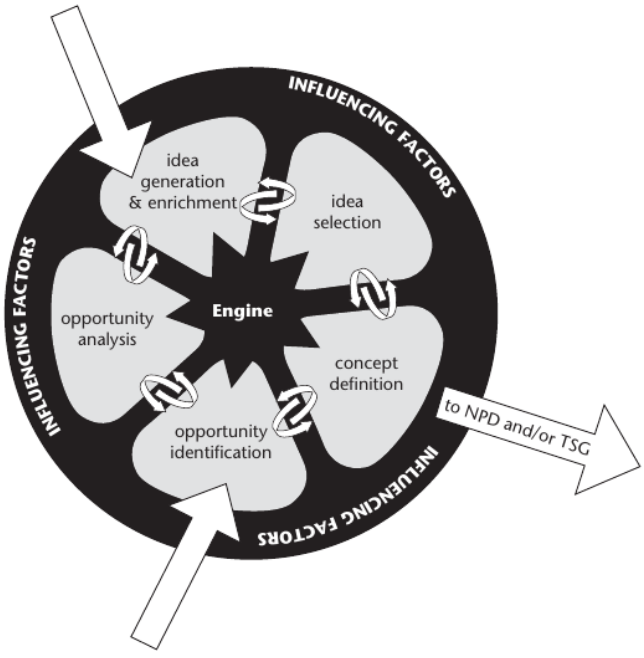

Fig. 4.1: NCD [2]

4.1.1 The Engine

"The element of leadership, culture, and business strategy sets the environment for successful innovation" [4] and it's in this environment that the main approach factors to the problem are centralized.

4.1.2 Five Elements of the Activity

The NCD identifies "five key elements that are controllable by the corporation" [4]. In the next sections, those elements will be clarified and adapted to the project team perspective.

Opportunity Identification

Identifying opportunities that organizations might want to pursue it's the aim of this element. The identified goals are "tipically driven by the business goals" [4].

Common industry organizations usually have defined hierarchic structures that significantly vary from software industry structures. Although the procedures used in the common industry are not similar to the software industry, why can't common industry benefit from agile values and principles?

This project opportunity is a consequence of common industry medium and small companies work-style.

In a general perspective, common industry medium and small companies are:

- **Unorganized**
- **Hour based work**
- **Product oriented**
- **Hierarchic dependent**
- **Unmotivated**

The identified opportunity aims to allow common industry perspective to work with an agile, client oriented perspective and consistently improve work flaws over time.

Opportunity Analysis

"Opportunity is assessed to confirm that it is worth pursuing" [4].

To accomplish this goal, a study was conducted in four different companies. These companies were chosen due to their natural flow, objectives and areas being very different from each other.

ALopes Industry is a company that represents Mahlo and Mathis machines in Portugal. Either Mahlo and Mathis are companies that focus it's strengths in quality control systems for the textile industry. ALopes Industry is usually focused in on site repairs and periodic calibrations. Based on a daily planning approach, tasks dependencies and unpredictable events that can require priority in the backlog, ALopes Industry requires very specialized planning skills.

ALopes Studio is a new IT Team that are focused in software, design and marketing. ALopes Studio develops own software and clients own projects. Composed by two collaborators and working in several simultaneous projects, ALopes Studio faces planning barriers using standard agile methodologies.

Fabsgarage is focused car projects and repairs. Counts with X collaborators and SOME MORE INFO.

FAP is a company that focus its resources in handmade furniture regarding client's own projects. Using a dynamic work style, FAP is able to build different products in the same production line.with a very talented and specialized team of eight collaborators.

The chosen organizations defend different values and work style. However, all of them can be approached with a client oriented perspective. This study was based in observation and feedback, in order to create a complete flaw scenario to be used in the concept definition and platform development.

Ideas Generation and Enrichment

By definition "idea creation and enrichment concerns the birth, development and maturation of a concrete idea" [4]. The core idea is to efficiently apply agile methods in the common industry with the help of specifically designed platform.

The idea generation and enrichment grew with the work process experiencing. Once facing unwanted scenarios in real task's, the concept development grew as much as the scenarios appeared. This gave the project a strong starting point, which became stronger upon the creation of hypothetical scenarios that helped prevent real difficult events.

Idea Selection

"The problem, for most businesses is in selecting which ideas to pursue in order to achieve the most business value. Making a good selection is critical to the future health and success of the business" [4]. Here, businesses aim to select what to pursue.

In this project case, there wasn't a need to choose in what to invest but to find the best way to create the concept. The solution was to let the concept evolve naturally and adjust to new needs over-time.

This implied to develop a modular standard concept and a modular standard and dynamic platform that was able to be adapted with a standard approach. Although this can be a confusing description, it will be explained in detail in the platform's Design chapter.

Concept Definition

Being the final element of NCD, here it's essential for the "innovator to make a compelling case for investment in the business or technology proposition" and justify the reasons of the selected approach.

This is project represents an intrinsic need for common industry companies, and will be a benefit for improving the work processes and work tools when comprehended by users and applied with the required company adaption. It will also allow common industry companies to benefit from agile methods repercussions

4.2 Value For The Costumer

The value of a product has always three main perspectives.

It's real **value**, usually defined by the product strengths and it's particularities that distinguish it from other similar products.

The **value for the costumer**, or the advantages that individual costumers will gain when using the product. This value concept is the difference between costumer gains and costs when acquiring or using the product.

And the **perceived value**, that represents the worth of a product in the mind of a costumer.

Following the previous definitions, the created agile concept, along with the specifically developed platform, supports the work process management to be capable of improving discipline and promote agility, in a planned and controlled work environment.

For the costumer, this value is not always straight. The existence of barriers is always contributing for the value reduction. The is existence of skeptical mentalities is very usual

in the project costumer segment and this consists in a huge contribution to the use of the concept. As a consequence, the concept adaption time can be longer than predicted and can increase the concept cost of usage.

Once the adaption phase is finalized, costumers can start to deeply explore the gains of the concept. The main value to extract, in the mind of the costumer, is the improvement of the work process quality, control and measurement, and consequently, the increased quality of their products and services.

4.3 Value Proposition

Objectively, this project is focused in medium and small common industry organizations and it purposes to create an agile framework for them. Allowing the centralization of IT Tools and information, such as planning requirements and decision making.

4.4 Canvas Model

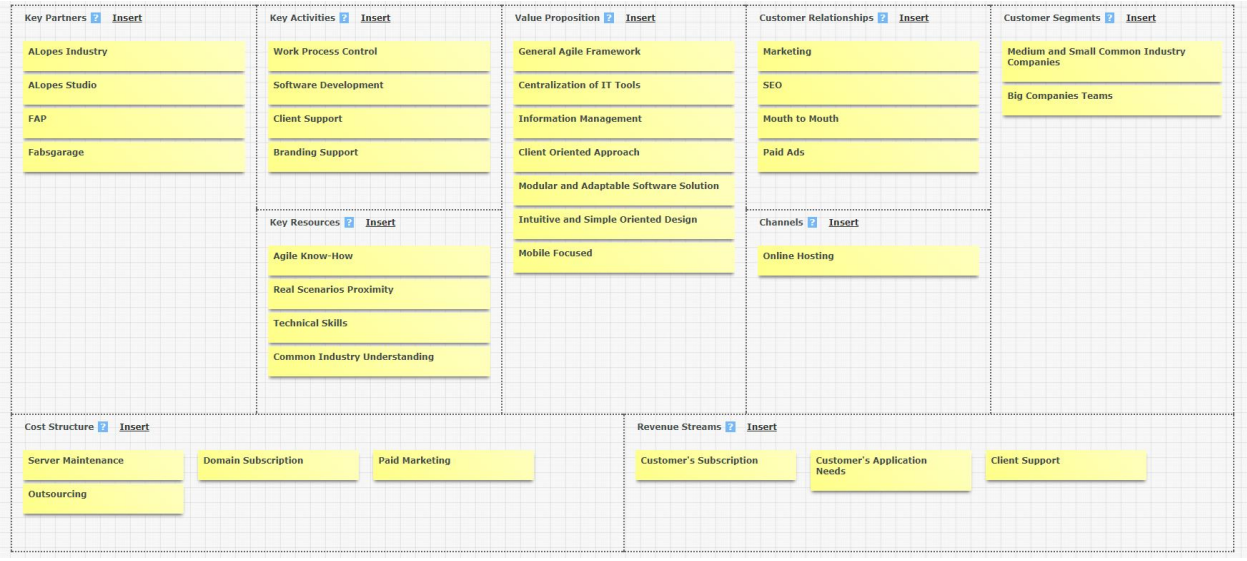

Fig. 4.2: Canvas Model

It is important to reference the costumer segments of the projects. Besides being two different segments, both are extremely similar in terms of work process. However, due to the different environment and needs, it's required to approach them differently due to

their intrinsic requirements. Medium and small common industry organizations and big companies teams are the focus of the project.

It is in this segment that the project aims to implement an agile framework to assist the work process and increase organizations efficiency.

4.5 AHP

AHP is a multi-criteria decision method that was developed by Saaty in 1976. And, "because of its well-known advantages, this method has been used successfully in various fields" [8].

For this demonstration, an online software called **easyHP** was used. This software was developed by Fernndez Martn, Daniel in 2016, and aims to automate the AHP method demonstrations [6].

The alternatives used in this method were not presented yet. For that reason, the following table is responsible to summarize the chosen alternatives.

Developing a specifically designed software	This alternative is a valid approach to the problem for each specific case. However, it would require a new software for each business to work with, no matter how good and versatile they are.
Adapt the work process to an existent platform and methodology	This approach is used by several organizations. The adaption usually has unwanted consequences for the work process to fit in the methodology that is, normally, bad implemented.
Create a new agile concept and a standard platform	This approach inherits design challenges that are difficult to overcome when creating a standard solution. However, this solution allows the usage of a modular adaptable concept that can be improved in a standard perspective.

Table 4.1: AHP Alternatives

Using the **client orientation**, **adaptability** and **standardization** as a criteria for the AHP method. The first question to ask, in order to define the criteria importance is, "which criteria is more important to you" [6]?

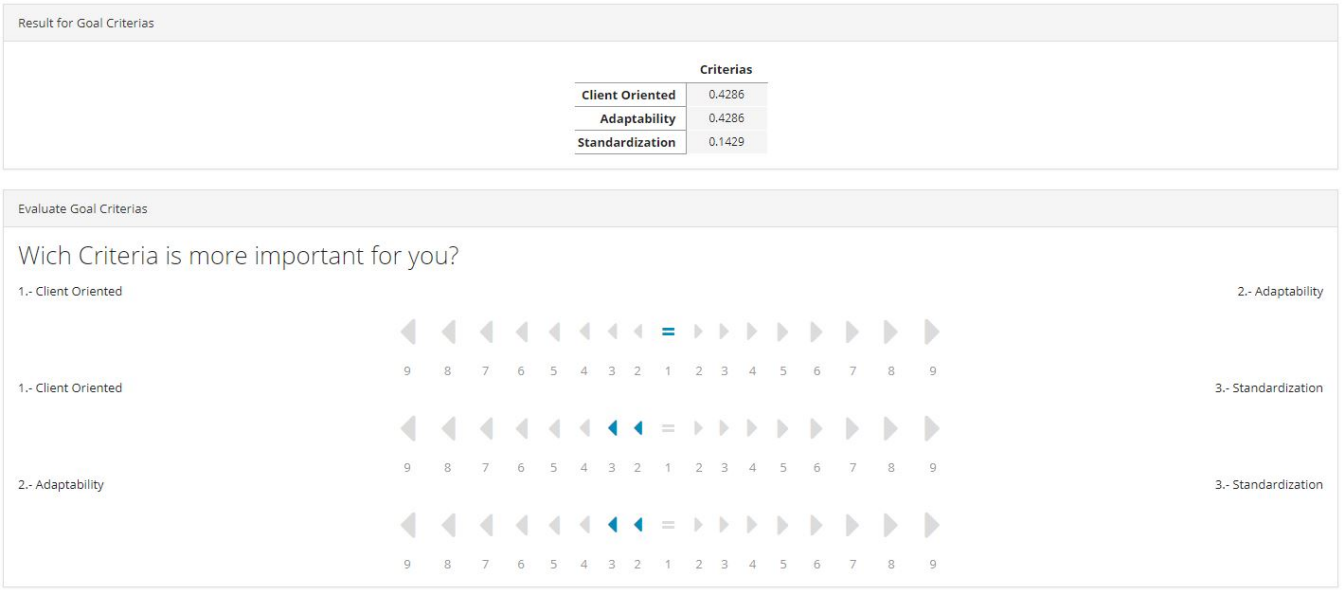

Fig. 4.3: Which criteria is more important? [6]

It is possible to verify that the importance of client oriented and adaptability criteria importance are higher than standardization. It's clear that organizations want to improve themselves first but, it's part of the project goal to standardize solutions and improve the common industry process.

Secondly, to define the criteria result per alternative, it's important to compare each alternative pair with the AHP scale.

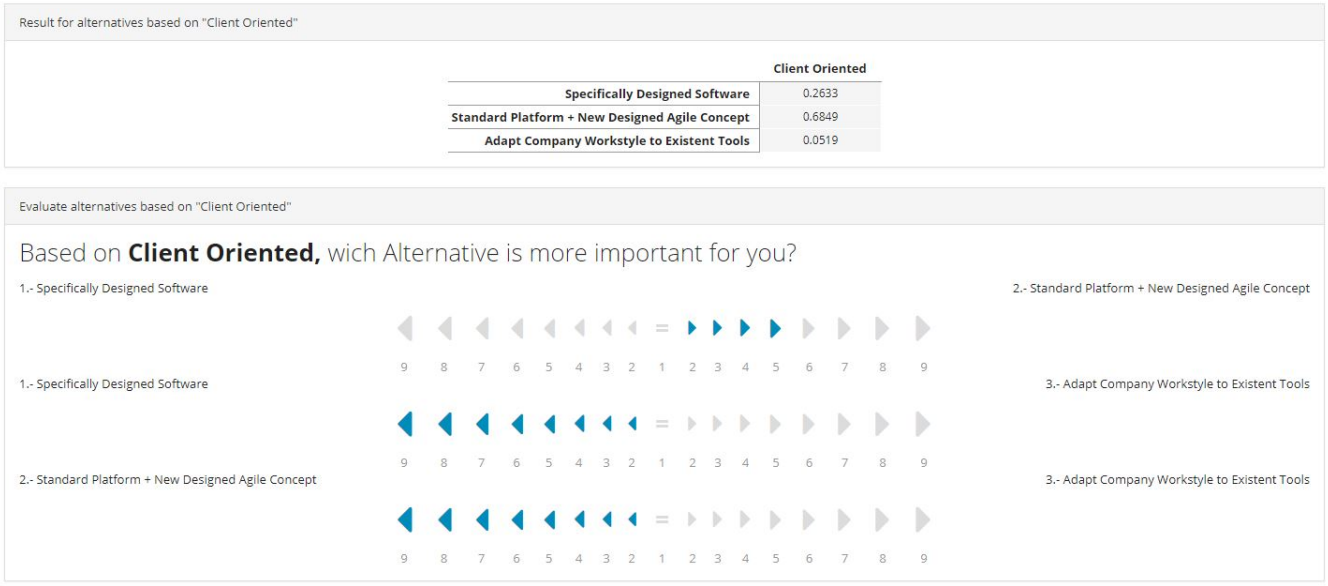

Fig. 4.4: Client Orientation Alternatives Evaluation [6]

In the client perspective, it's better to have the support and maintenance of a global software than a specifically designed software. This will reduce organizations software costs and improve their confidence in a long-term sustainable software.

In the other hand, a specifically designed software is more likely to succeed than an adapted platform which will force a work style change in the company.

The same perspective is applied to the project's own goal. It's always difficult to force existent methodologies into a work process.

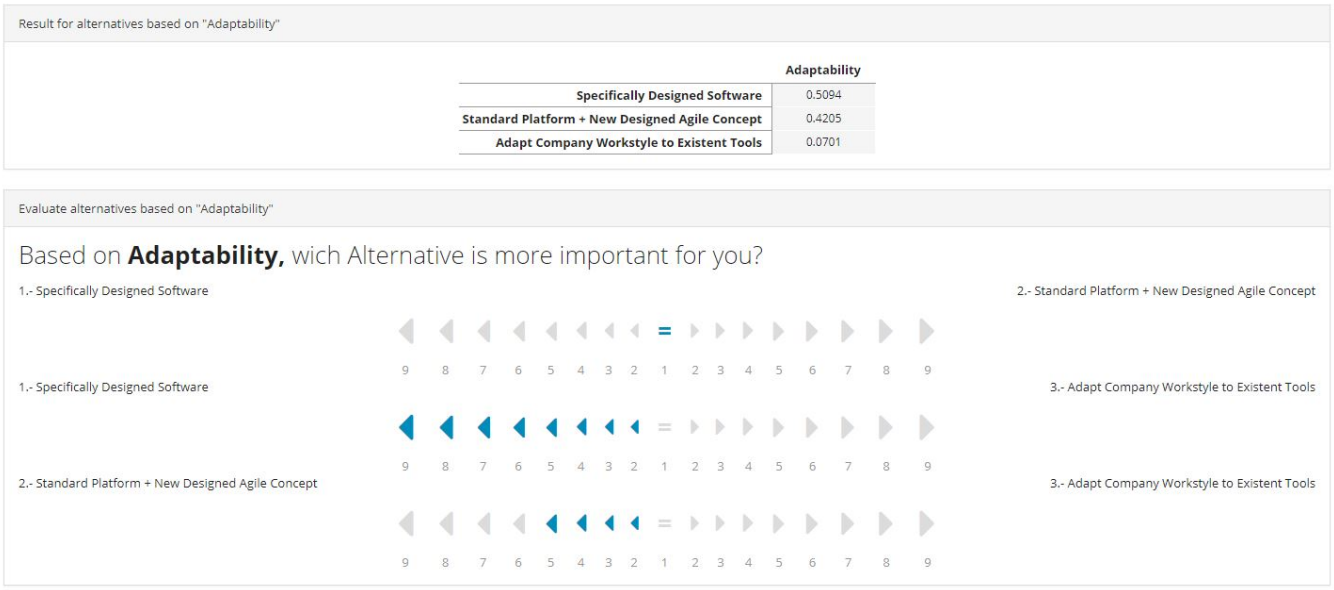

Fig. 4.5: Adaptability Alternatives Evaluation [6]

The adaptability criteria justifies that, for the client, there are no difference in having a specifically designed software or a standard platform with a new designed agile concept. Organizations need for adaptions will exist with either one or another and once they would be equally prepared to receive new requisites, there are no differences in this criteria, on this pair.

Once comparing a specifically designed software approach to the adaption oft he work style to an existent tool, it's possible to estimate that companies would be more likely to choose the first option, in order to avoid unwanted consequences.

The same perspective is observable when comparing the project goal to the work style adaption upon the implementation of existent tools. However, the preference difference over a specifically designed software is justified with the needs of adaption for each case in the project goal. Because, even being prepared to be adapted, it will always require a study of the organization to decide how that specific organization should use the method.

Although this topic may seem confusing, the upcoming chapters that are responsible of describing the product will clarify it with the solutions description and design.

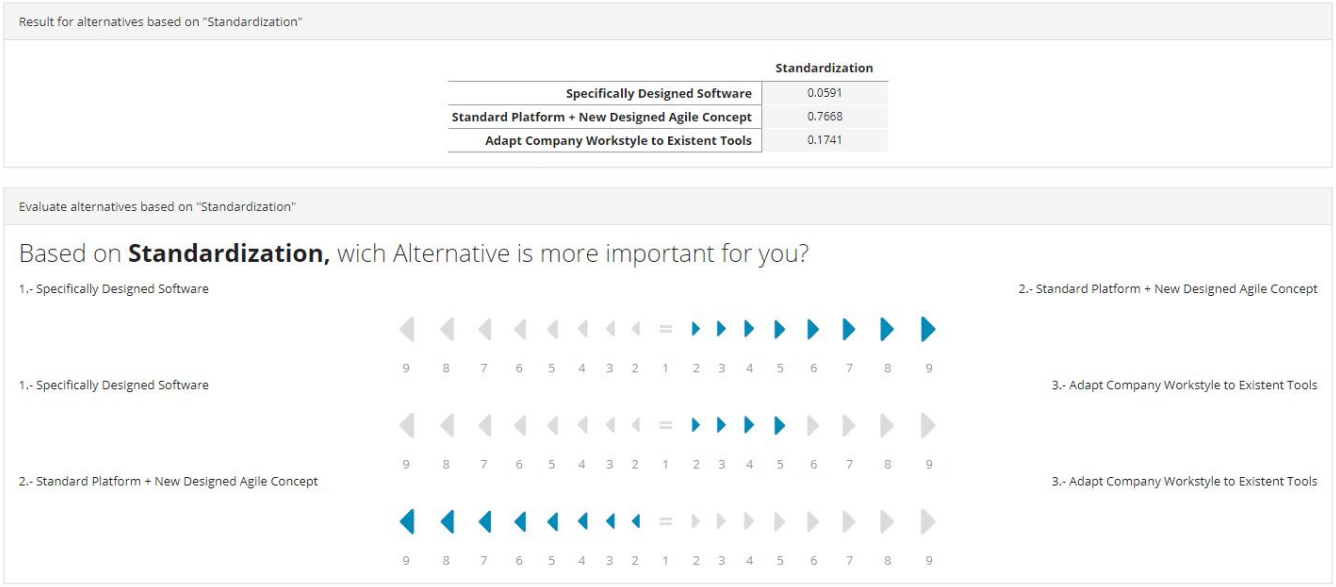

Fig. 4.6: Standardization Alternatives Evaluation [6]

Regarding a standardization perspective, the preference tends over the use of existent tools and the new designed agile concept.

The attribution of the work style adaption evaluation, when comparing to a specifically designed software, is not higher due to not solving companies needs. However standard, this tools won't be able to healthily maintain a sustainable work process.

This last explanation also justifies the evaluation of the project goal, when comparing to the usage of different tools. However both are standard solutions, only the project goal is adjusted to the common industry. This makes the use of standard solutions not a standard to the common industry.

Finally, the result of the AHP method, with the given evaluation supports the project approach, as we can see in the next image.

Survey Result

	Client Oriented	Adaptability	Standardization	
	0.43	0.43	0.14	RESULT
Specifically Designed Software	0.26	0.51	0.06	0.34
Standard Platform + New Designed Agile Concept	0.68	0.42	0.77	0.58
Adapt Company Workstyle to Existent Tools	0.05	0.07	0.17	0.08

Fig. 4.7: AHP Result [6]

As it is possible to verify, the project goal has the higher result with 0.58 over the provided alternatives.

Sprint Task Methods

5.1 Methodology Concepts

As stated in the third chapter, agile is not a method but a mindset that values "individuals and interactions over processes and tools" [3].

Since the introduction of SCRUM and Kanban into the software industry, software development quality has been improving drastically [10], because these methodologies are able to create a natural work environment inside the software industry.

However, applying current agile methods in a non-software environment is not sustainable and may turn the work process inefficient. For that reason, the work-flow flaws experienced in the common industry were observed in order to create an agile method that could provide strategic solutions to the most common and specific problems in a company.

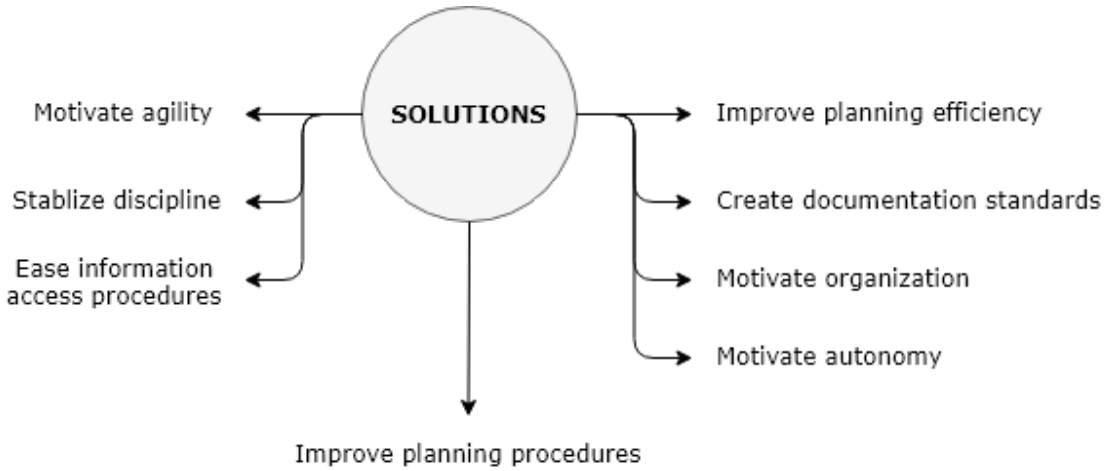

Fig. 5.1: Sprint Task methods solutions

In order to understand the platform decisions, it is important to ask how will Sprint Task accomplish these solutions and what impacts will they have in the work-process?

Before answering the previous questions, it is important to clarify some concepts that may have different meanings considering different scenarios and can compromise the understanding of Sprint Task methods. The following table aims to objectively define a group of important Sprint Task concepts.

Sprint	A sprint is a time-line gap where tasks are executed. A sprint is defined by a starting date and a sprint type. Collaborators see sprints as goals and objectives, where they can work on their assigned tasks with all the required tools. Before a sprint starts, the manager will plan it by assigning backlog tasks to teams and, at the ending of the sprint will a retrospective review will also be the managers responsibility.
Project	Common industry projects differ from software industry projects, not only in technical aspects but also in its own characteristics. In a non-software environment projects are usually simpler and their duration is usually shorter. Having teams working in several projects at once is a characteristic of the common industry and contrasts with the usual software development approach in the software industry where this does not usually happen.
Task	A task is a project need or requirement. Tasks can only be created and assigned by managers and executed by collaborators.
Operation	Created by collaborators, an operation is a smaller task that belongs to a parent task. Operations aim to split bigger tasks into smaller ones and improve the documentation of collaborators procedures. Also, operations will be part of retrospective artifacts, such as tasks and projects summaries.
Application	To solve the need for organizations specific tools, Sprint Task platform will work as a multiple standalone project container. The main application will support Sprint Task methods and allow users to access an application market with general and custom made applications defined by the company. In other words, every organization will have access to the standard platform and general applications. However, organizations with special requirements will have the possibility to purchase special tools and request the development of new ones. Applications are be standalone projects that can be integrated with tasks, projects, operations and other applications.
Retrospective	A retrospective review occurs at the end of a sprint or a group of sprints and aims to allow managers reviewing "Done" tasks and improve work-flow issues on their teams.

Table 5.1: Concepts Contextualization - part 1

Team	A team is a group of one or more collaborators. One collaborator can be part of multiple teams.
Sprint Type	A Sprint Type is defined by a name, and a duration. Sprint Types can be either active or inactive.
Backlog	It is part of the backlog, any task that is not assigned to a team in a current or upcoming sprint.
Active Sprint	Usually agile frameworks use sprints as a constant interval project concept but, Sprint Task defines them as work process requirement instead. In fact, Sprint Task allows multiple sprints to coexist in the same time interval and share the same projects over the same teams. Visually, the active sprint is seen as a common sprint however, it is a context filter of coexistent sprints, projects, tasks and teams.

Table 5.2: Concepts Contextualization - part 2

5.2 A New Agile Concept

Inspired by Scrum Nexus and Kanban, the new agile concept aims to create efficient solutions for common problems experienced in common industry work-flows. In order to describe the conceptual solution in a technical perspective, the domain model was separated to create an overview over the relationship of projects and tasks, the active sprint, the retrospective, the application and form concepts.

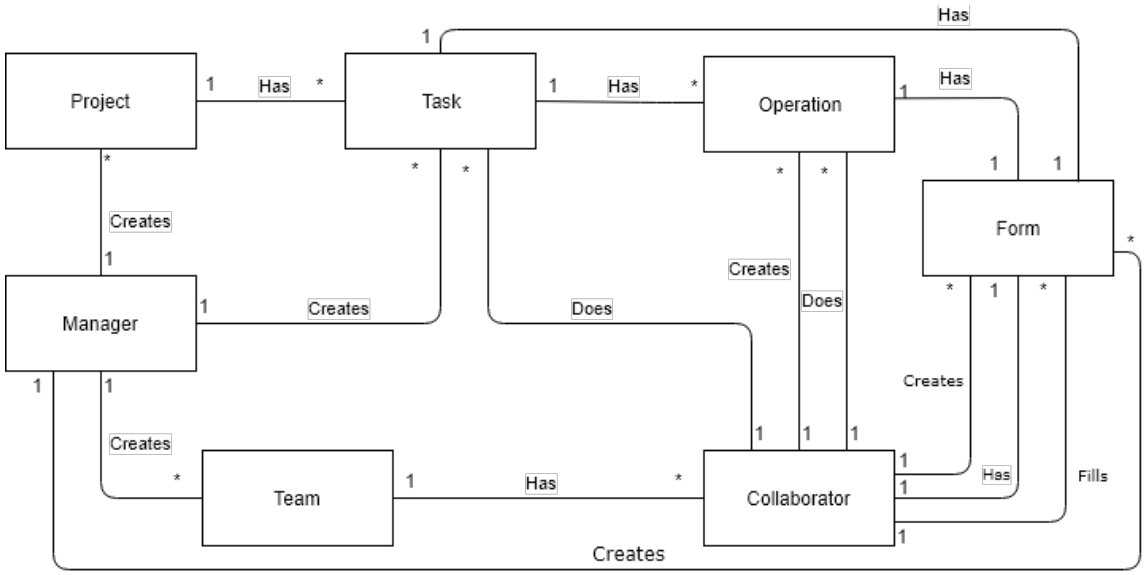

Fig. 5.2: Projects and Tasks Relationship

In order to highlight the existent relationship between projects, tasks, teams and operations, the previous diagram defines the standard behaviour of managers and collaborators in a Sprint Task context.

Managers are in a higher position than collaborators in the Sprint Task hierarchic structure, and they are responsible for preparing teams, in order to assign previously created project tasks, in a regular basis.

Although, collaborators are only responsible for executing tasks, they are also able to create operations to allow managers to access a complete documentation of procedures, when tasks are not representative.

As stated in figure 6.2, a project is created by managers and can have multiple associated tasks, also created by those. Tasks are executed by collaborators and can have multiple associated operations, created and executed by these ones.

Apart from the described relationships and dependencies, the concept of form appears in the diagram has an artifact associated with tasks and operations. It was represented in the previous figure in order to reference external applications from Sprint Task application market. Forms are customizable sets of questions, procedures and checklists that can be created by managers and associated to tasks and operations. Either managers and collaborators are able to associate forms to tasks and operations, and export them separately or with tasks, operations or sprints.

In Sprint Task, multiple sprints can occur simultaneously and share teams, projects and tasks. The sprint type concept was introduced to ease the sprint management procedures, such as planning and reviewing. A sprint type can either be active or inactive, and only active sprint types will result in sprints.

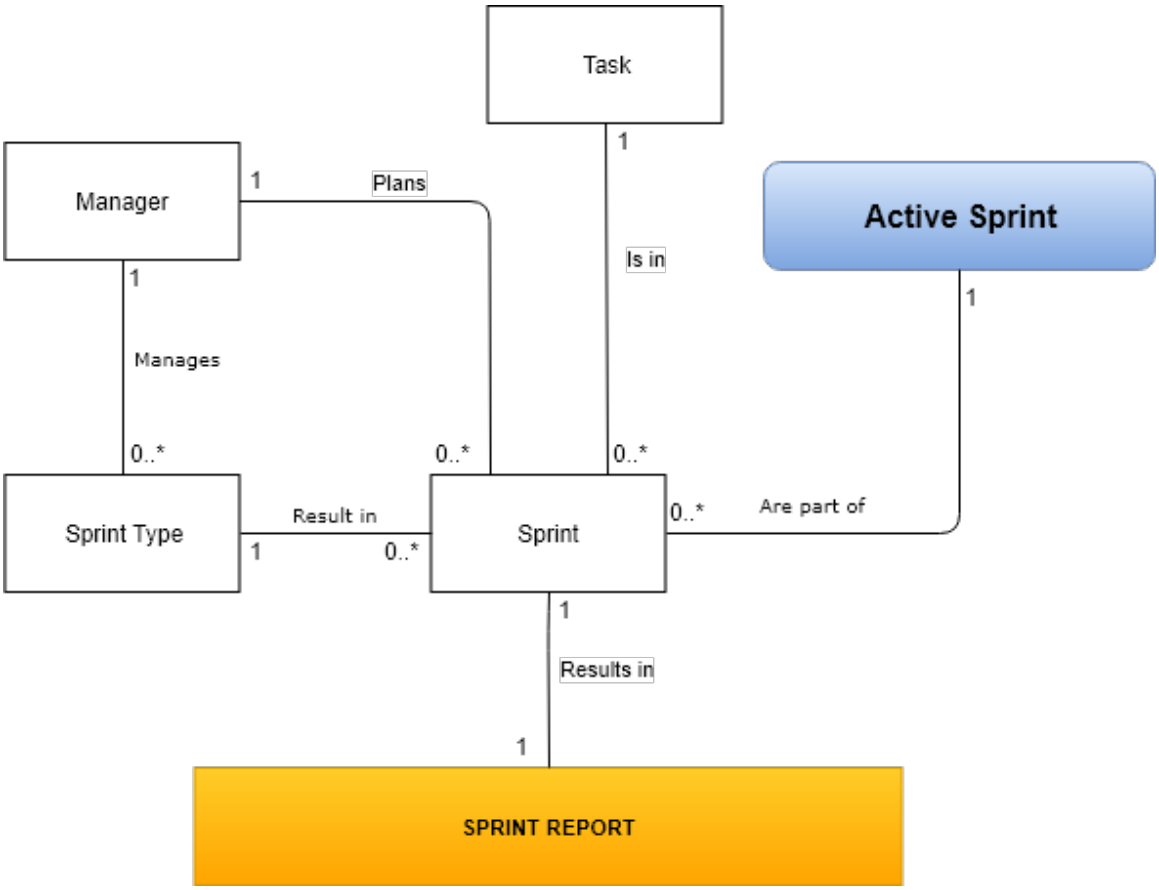

Fig. 5.3: Active Sprint Concept

On a daily basis, sprints will try to renew themselves according with the active sprint types. If a sprint type is renewed, the current associated sprint is transformed into an outdated sprint, the upcoming associated sprint is transformed into a current sprint and a new upcoming associated sprint is created.

The active sprint is directly associated with the current sprints, it wraps all of the current sprints associated artifacts and provides automated procedures to filter and contextualize the active sprint information, according to each situation.

At the end of a sprint, recently executed tasks are added to the review process, so that managers can accept or decline. In this last case, tasks are re added to the backlog.

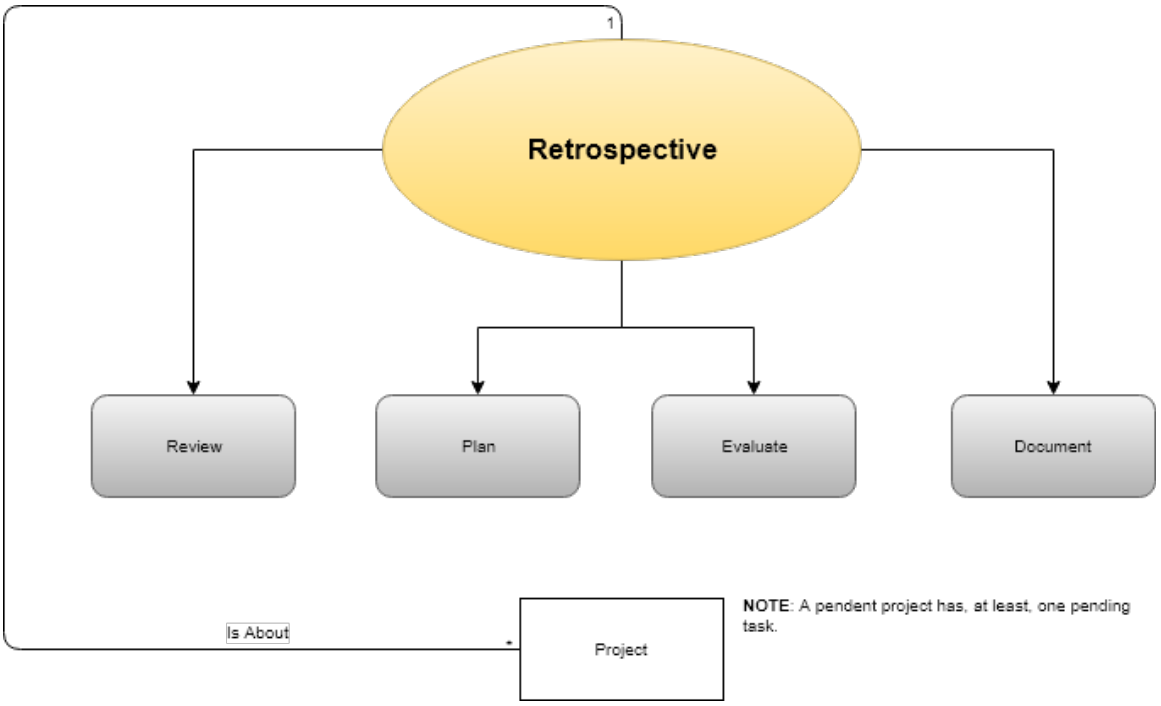

Fig. 5.4: Retrospective Concept

Ideally, a sprint review process shall result in a retrospective analysis, where managers can improve inefficient procedures or practices within the work-flow.
During the retrospective analysis, managers are motivated to discuss negative work-process points with collaborators. Also, they able to review sprints, plan upcoming sprints, evaluate the work-flow and document sprints with Sprint Task tools.

Approaching the application structure, it is possible to understand the role that independent applications assume in a company work-process. As stated before, the centralization of external applications in the application market, allows them to optionally be integrated with Sprint Task platform. The following diagram describes the relationship between independent applications and the Sprint Task platform.

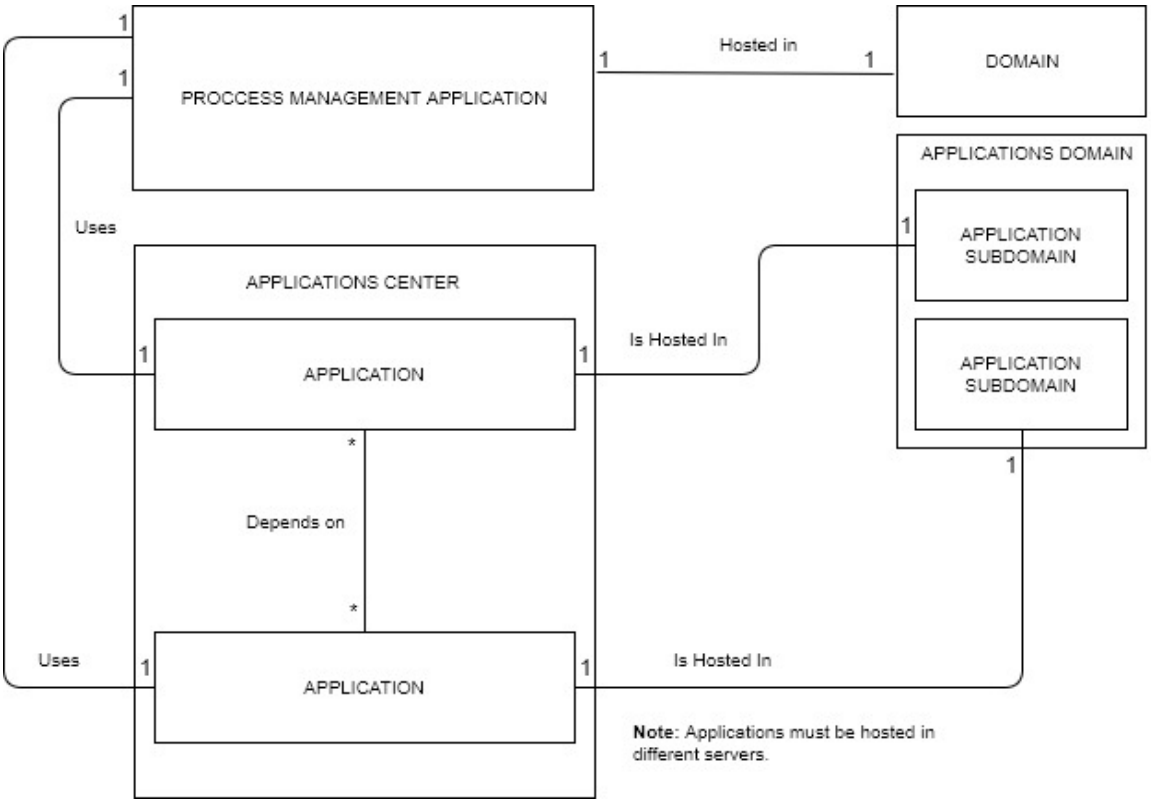

Fig. 5.5: Application Concept

Applications are hosted in different domains and different servers from Sprint Task platform, in order to provide independence between each application and the platform. They are meant to integrate Sprint Task platform, by associating applications connections and artifacts to tasks and operations.

There are situations where external applications can interfere in Sprint Task user interface. In these cases, the application impact and implementation may be studied in order to create the most sustainable solution.

However, applications can also work as separated software tools in an unsynchronized version of the application.

Detailing an application example, the following diagram describes the form concept used in the forms application.

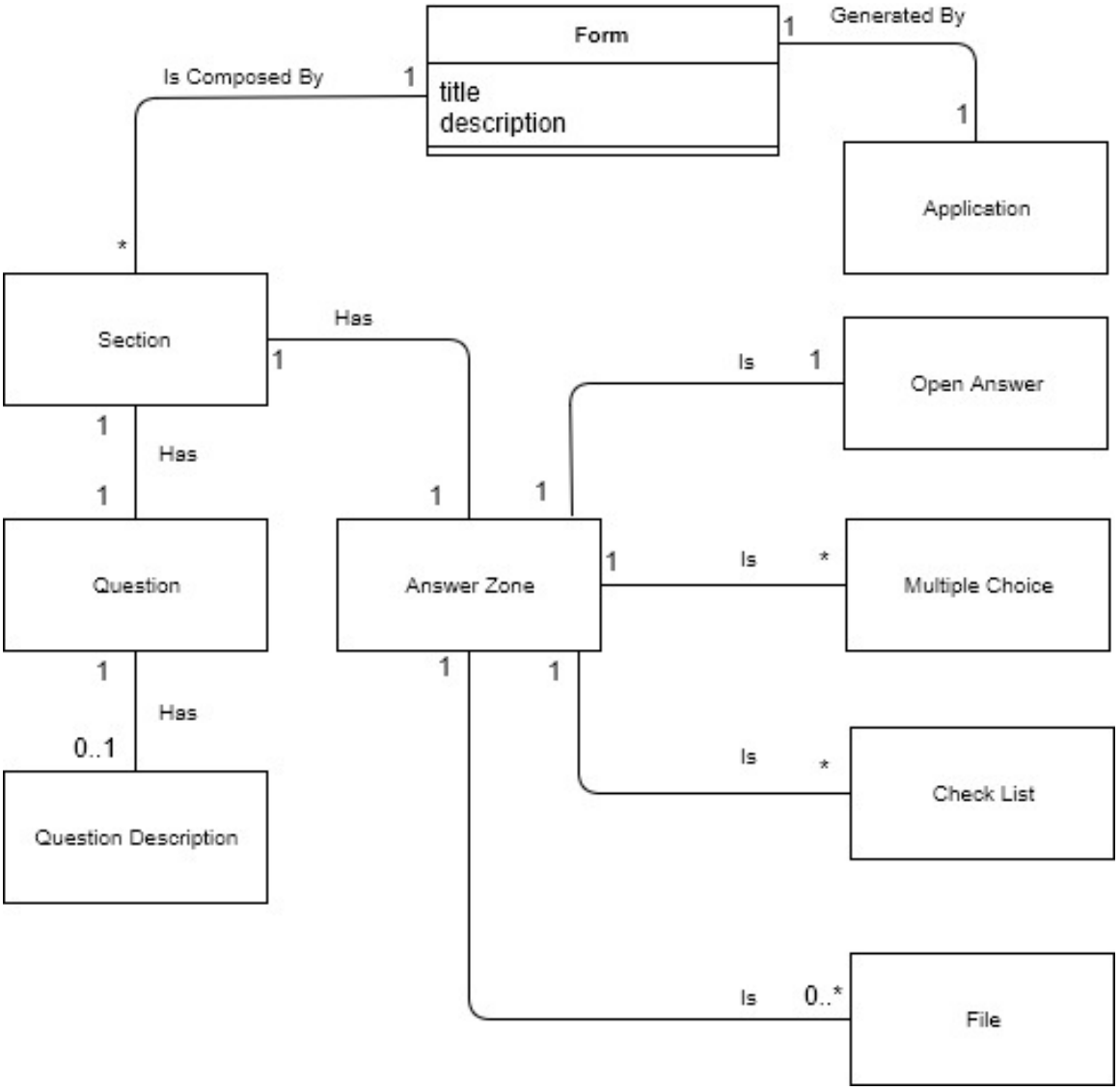

Fig. 5.6: Form Concept

As the figure 5.6 suggests, the form concept is independent from Sprint Task. This applications aims to allow users to build and create modular forms, with open answers, file uploads answers, multiple choice and selection questions.

In order to manage permissions, applications will use decentralized authentication services to allow and validate user actions.

5.3 Methodology Flow

Sprint Task does not demand a sequence of procedures. However, it suggests a guideline that defines the best practices that work-flows should follow in order to take maximum advantage of Sprint Task methods.

When an organization is adopting the referenced methodology, it is always required to study the impact of Sprint Task flow in the company core procedures, in order to increase efficiency and avoid unnecessary practices.

The following diagram aims to visually represent Sprint Task methods and clarify the relationship between teams and managers.

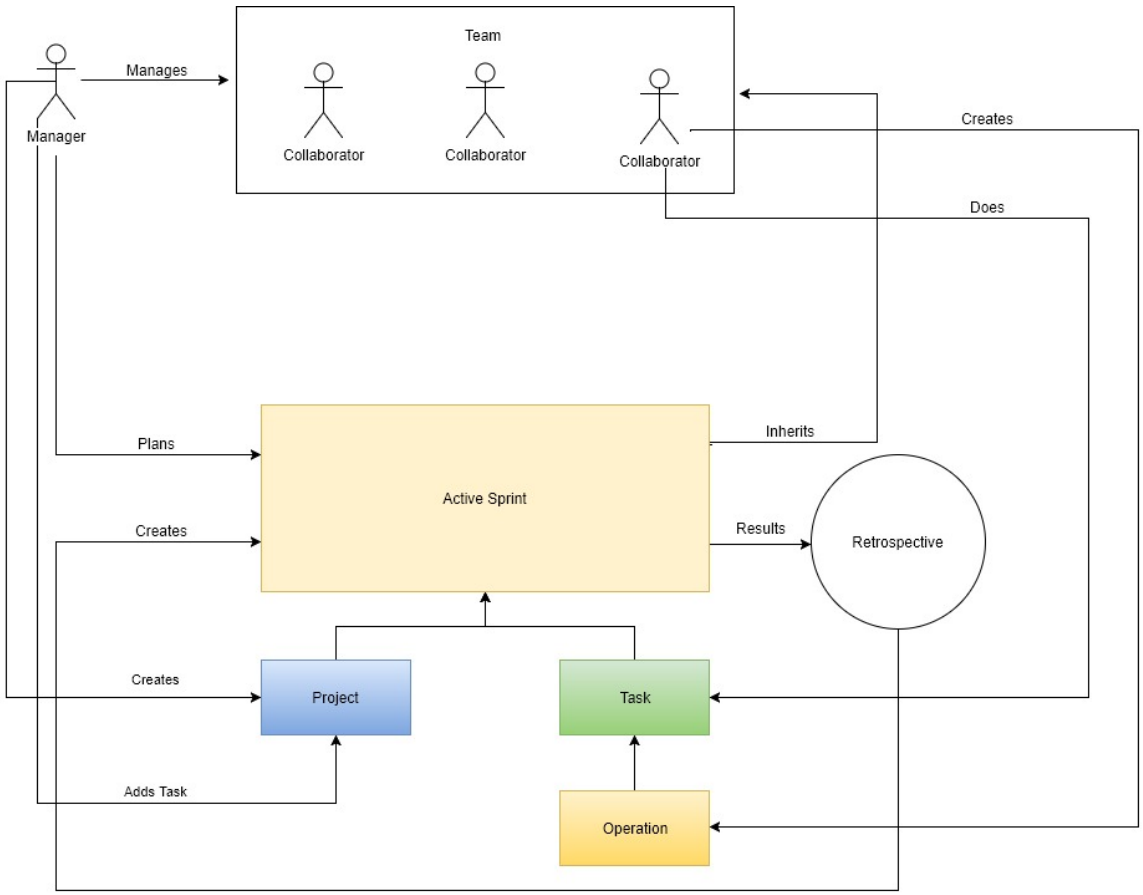

Fig. 5.7: Process Diagram

In a general Sprint Task overview, the manager role is responsible for planning, controlling and evaluating its teams performance. Also they are required to plan sprints, for maintaining a sustainable active sprint over time.

The active sprint can, in fact, be visualized regarding each different context needs, and this perspective allows Sprint Task methods to easily adapt in different organization environments without changing their managers planning mindset or collaborators access to information.

To manage the active sprint, a manager needs to manage sprint types according to the work-flow requirements. Active sprint types are checked in a sprint renewing event each day, and, if the conditions are reunited, a new sprint will be created and the active sprint is updated. Each time a sprint is created, its outdated referenced tasks marked as "Done" are included in the managers reviewing event, known as retrospective.

5.4 Platform Approach

Once designed the software platform, it's important to clarify each business concept to be implemented.

First, a Project is composed by several Tasks. This allows the project to be decomposed in smaller parts and to assign them to the responsible and specialized organization Teams. A project that is not considered "Done", may be planned or re-planned at anytime.

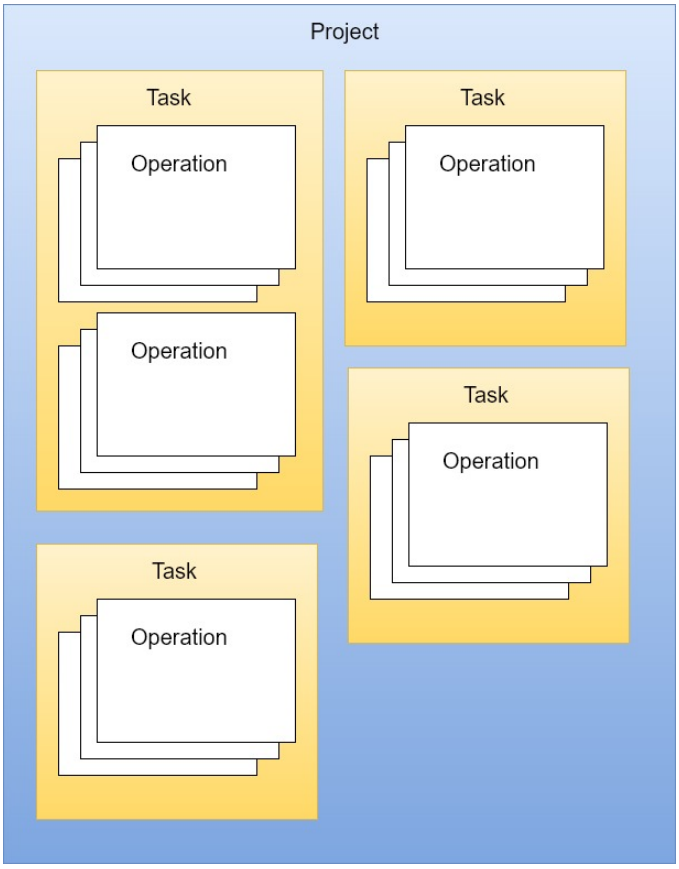

Fig. 5.8: Project Structure

Tasks were initially planned as a standalone concept with dependencies. Forms, Observations and Operations could have been part of the Task structure.

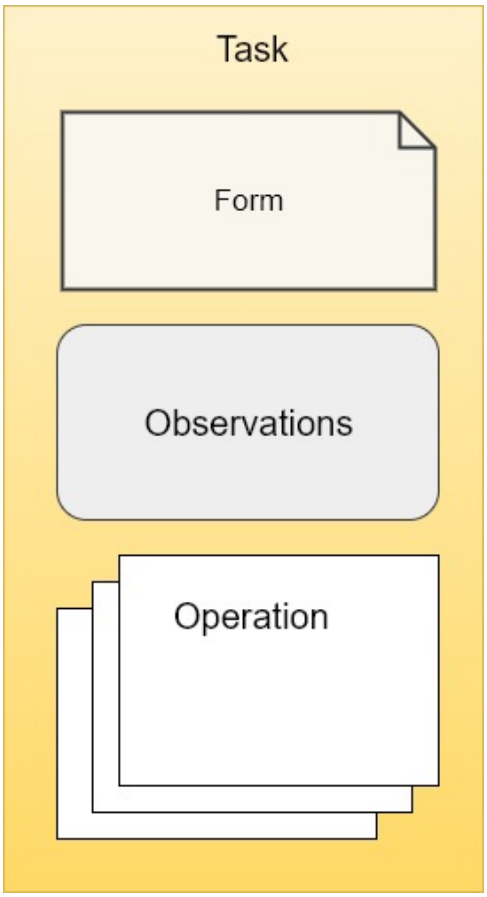

Fig. 5.9: Task Structure

Operations are also similar to tasks and were initially planned to integrate forms and Observations. As referenced before, Operations can only be created by collaborators and are meant to decompose big tasks.

Solution

6.1 Use Cases Overview

6.1.1 Global Overview

Sprint Task platform aims to support managers procedures on planning and sustaining multiple projects. Also, it provides important tools to collaborators, allowing them to preform better on their tasks.

With that mission, Sprint Task provides a vast variety of functionalities, as suggested in the following diagram.

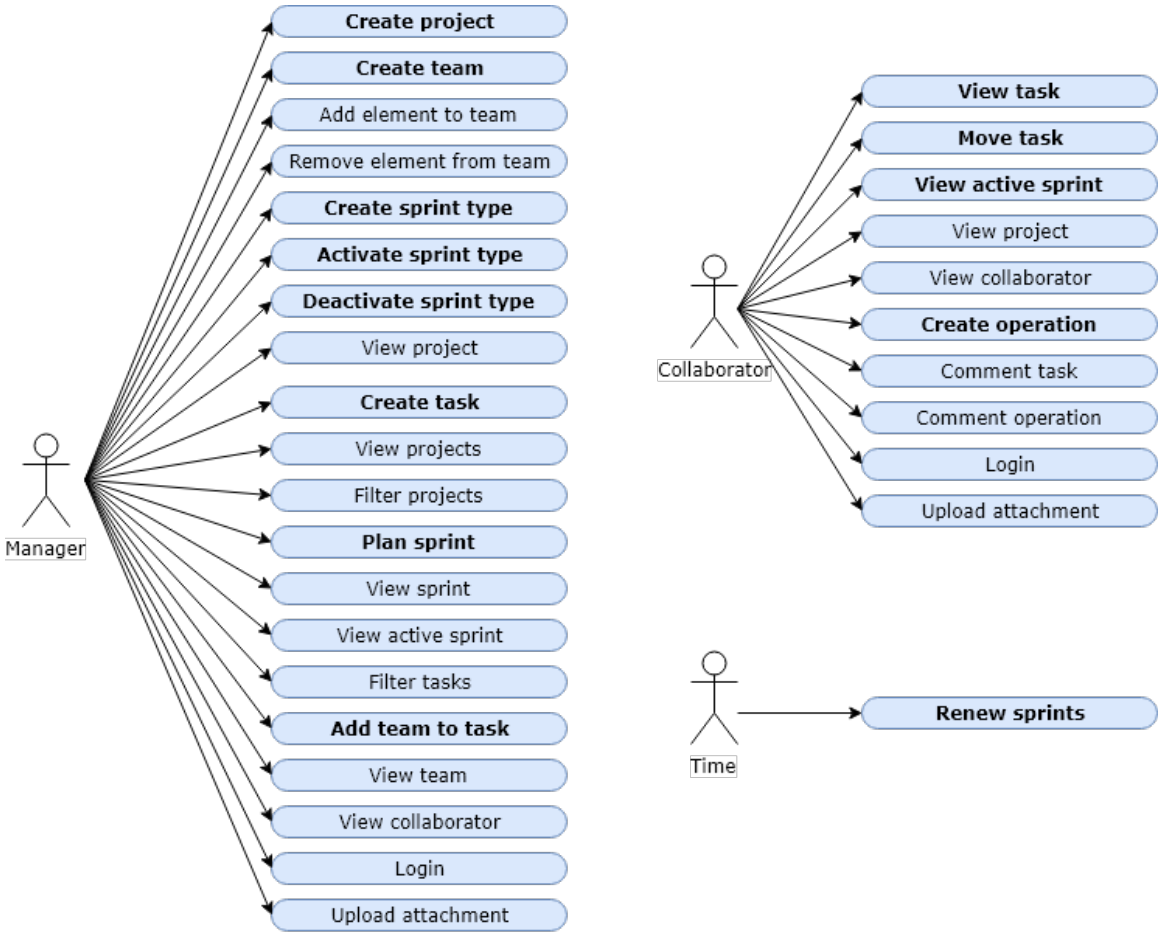

Fig. 6.1: Use Case Diagram

As suggested, Sprint Task has three important action triggers, time, managers and collaborators. Each one is functionally dependent of the two others.

Time is responsible for renewing sprints, in order to maintain a sustainable and organized working environment. This allows the work-flow to follow a pre-specified rules, procedures and artifact generation events.

Managers are responsible for planning the work-flow, taking in consideration their own work environment characteristics. They are able to create projects, tasks, teams, sprint types and manage each of these concepts.

Collaborators are responsible for guaranteeing the work process sustainability and communicating their results. Sprint Task strongly encourages them to be autonomous and independent, allowing collaborators to manage their own active sprint. Also, they are responsible for creating operations to allow a stronger task insight on managers.

6.1.2 Time

Time assumes a role that aims to maintain a sustainable and renewable work-flow chain, in which, each new chain link is a pre-defined time interval in Sprint Task platform and an important mark in companies goals list. On each sprint renewable event, the active sprint is affected and so the managers and collaborators perspective about it. The next diagram aims to describe how sprints renewed internally.

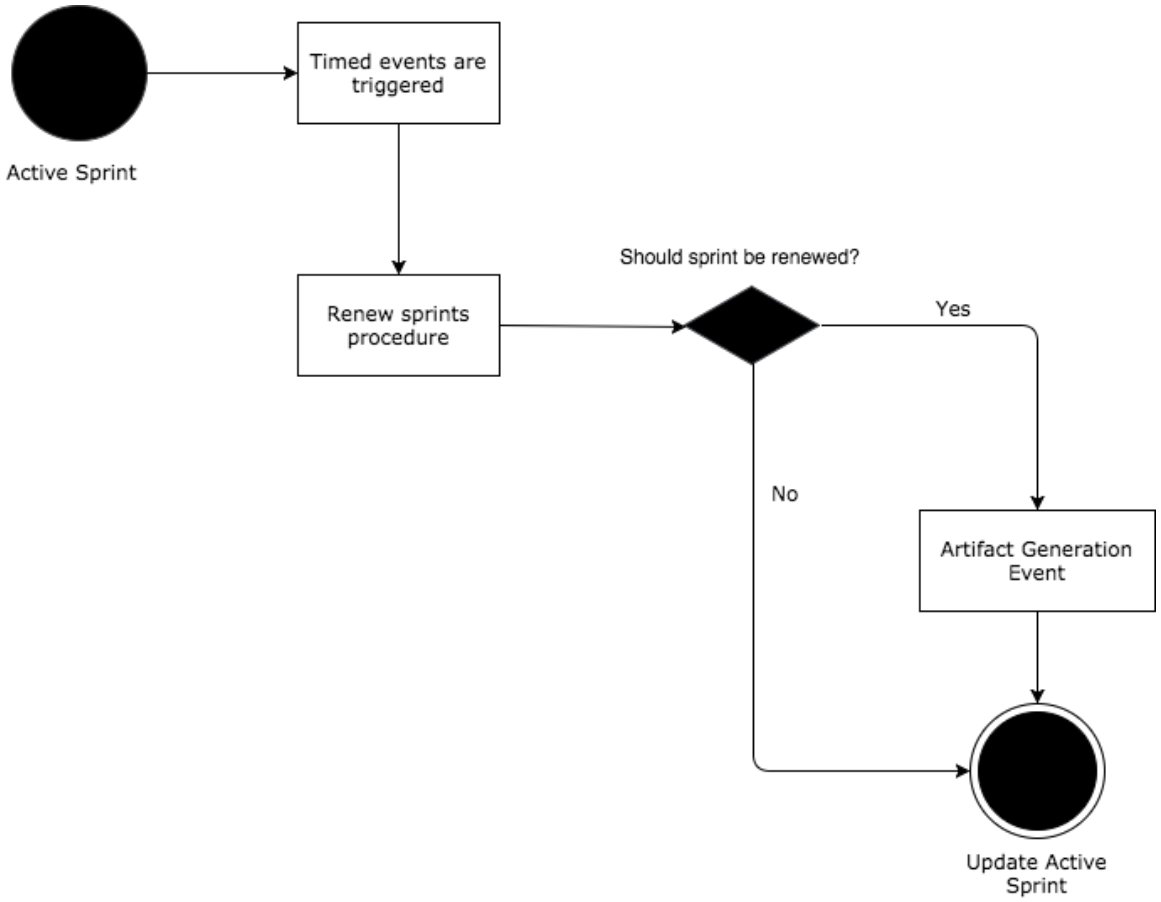

Fig. 6.2: Renew Sprints Procedure

The active sprint is an active filter that aims to return all the current artifacts for a given context of a given user at a given time. Everyday, at midnight, a sprint inspecting event is triggered and selects the sprints that need to be renewed. It will generate new active sprint content and add artifacts to the review process, according to the updates done to sprints. Also, the generation of automatic sprint artifacts that are available for download and share on the managers page, is triggered if a sprint is renewed.

6.1.3 Managers

For managers, tyhe importance of planning and delegate tasks is essential, independently of their teams working areas. It is part of their responsibilities to control and evaluate the quality of completed tasks and to fix issues in their teams before being critical and unsustainable. For these reasons, it is important to improve managers planning skills over time.

Managers use projects as their focus and teams as their assets. But Sprint Task allows them to use projects as assets and teams as their focus by creating a project-driven team oriented planning concept.

Sprint Task separates projects into tasks allowing different teams to operate simultaneously in the same project, while having real-time access to updated project artifacts and updates. With this approach, managers can plan several iterative projects in the same time interval and assign them to one or multiple teams. Also, Sprint Task is focused on teams efficiency and support, for that reason, it is prepared to prevent work overflow, promote team decisions and reducing hierarchic dependencies, by reducing the managers impact on teams work-flow. As a consequence, managers gain more control over the outcomes and process, in order to support their teams with appropriate material in the planning phase.

Concluding, managers are responsible for managing teams, control outcomes and respond to project deadlines on time. And, the manager role in the Sprint Task platform, also assigns core responsibilities to them, as it is displayed in the following diagram.

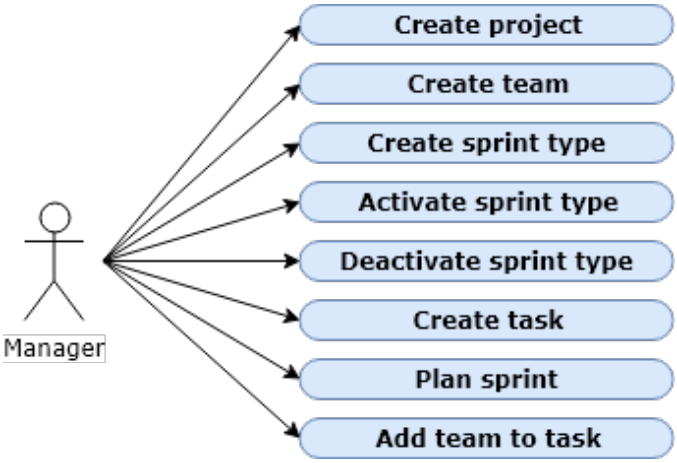

Fig. 6.3: Managers Core Responsibilities

6.1.4 Collaborators

Having a clear project insight and responsibilities overview is an important factor for an high efficiency rate in a work-flow. Sprint Task methods aim to increase collaborators impact and autonomy.

Regarding the relationship between Sprint Task and collaborators, their only responsibility is to execute and document work tasks.

However, Sprint Task allows them to be supported by core information from the company and managers planning activity.

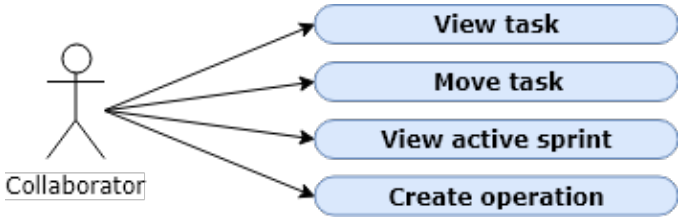

Fig. 6.4: Collaborators Core Responsibilities

6.2 Sprint Task Platform

6.2.1 Platform Architecture

Sprint Task was planned as a client-server web application based on restful services managed by a subscription plan. Focused on common industry work-styles, the user experience concept requires an easy and similar adaption to mobile devices, in order to simplify the access to the platform in specific day-to-day tasks.

The common industry needs are different over the areas. And, despite the need of planning, there might be areas that don't have anything in common. For that reason, in order to achieve a global internal problem solving method, Sprint Task includes an application market that aims to allow subscribers to order, buy and manage their applications. These applications will initially be developed by ALopes Studio, and will be focused in the clients feedback and needs.

Adopting a technical approach to describe the Sprint Task platform, the following diagram aims to clarify the platform architecture.

Since the beginning of the project, the development of the Sprint Task platform was fully based on emerging open-source technologies. Either the backend and the frontend was written in JavaScript, being the backend based on NodeJS and the frontend based on ReactJS. For persistence purposes, the analysis of the platform data-flow supported the usage of relational databases because the data transactions between client and servers were relatively small. For that reason, MySQL was chosen as the main database technology for the presented systems but it is not mandatory for applications, from the application market, to be based on MySQL.

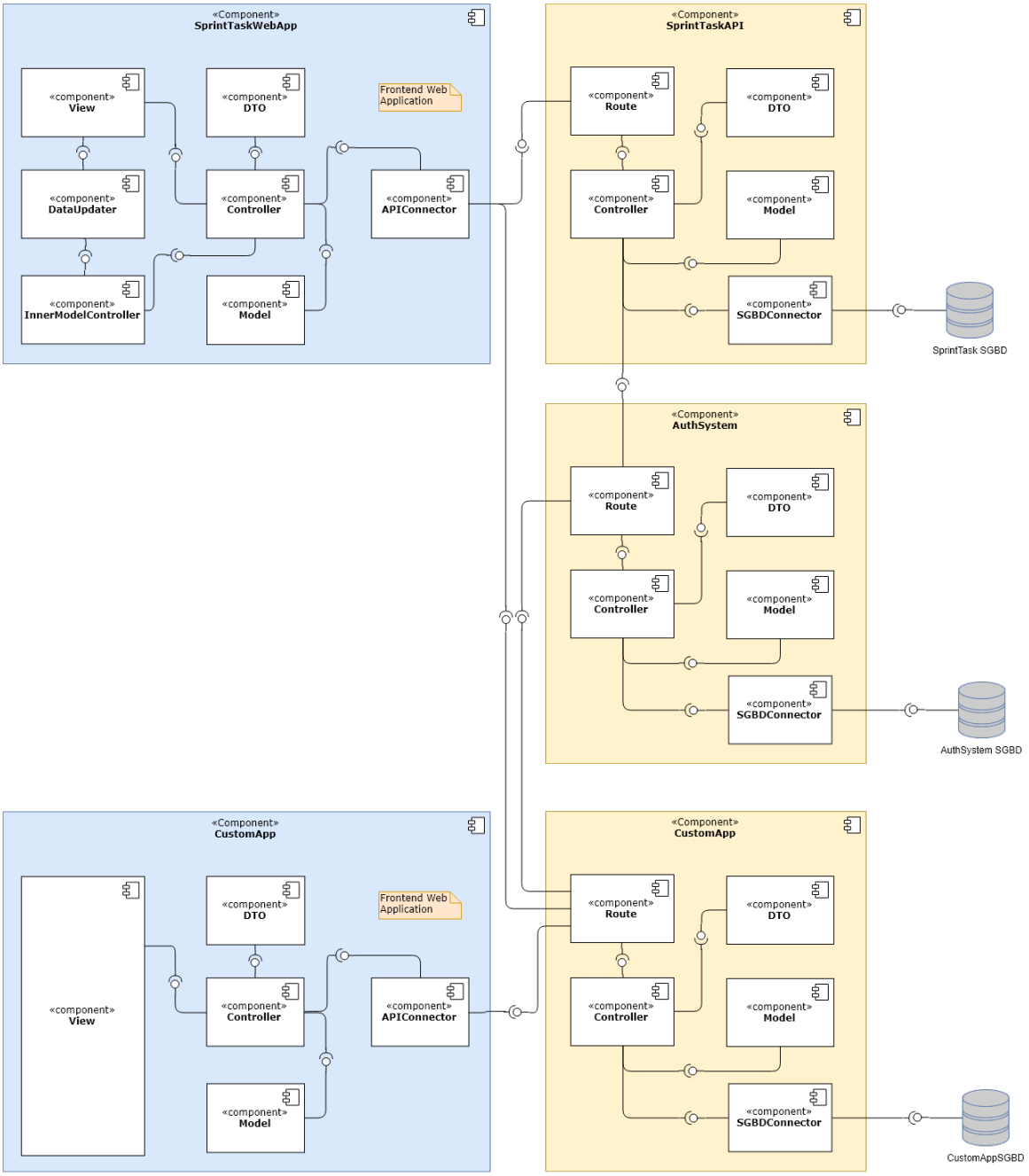

Fig. 6.5: Platform Architecture

The Sprint Task platform is composed by several independent synchronized systems working together at the same time. Due to the decentralization approach, the authentication services are used by each system and aim to provide user authentication and token validation services over time. Either Sprint Task main platform and applications from the application market depend on the Authentication System to authorize information accesses for users. With this approach, authentication operations will not affected other systems on the platform.

6.2.2 Authentication System

Aiming to provide decentralized authentication services to Sprint Task and its applications, the Authentication System is a web API based on NodeJS and MySQL that is responsible for gathering the platform users, manage their subscriptions and validate their actions according to each platform.

In order to validate users over time, the Authentication System uses the approach described by the following diagram.

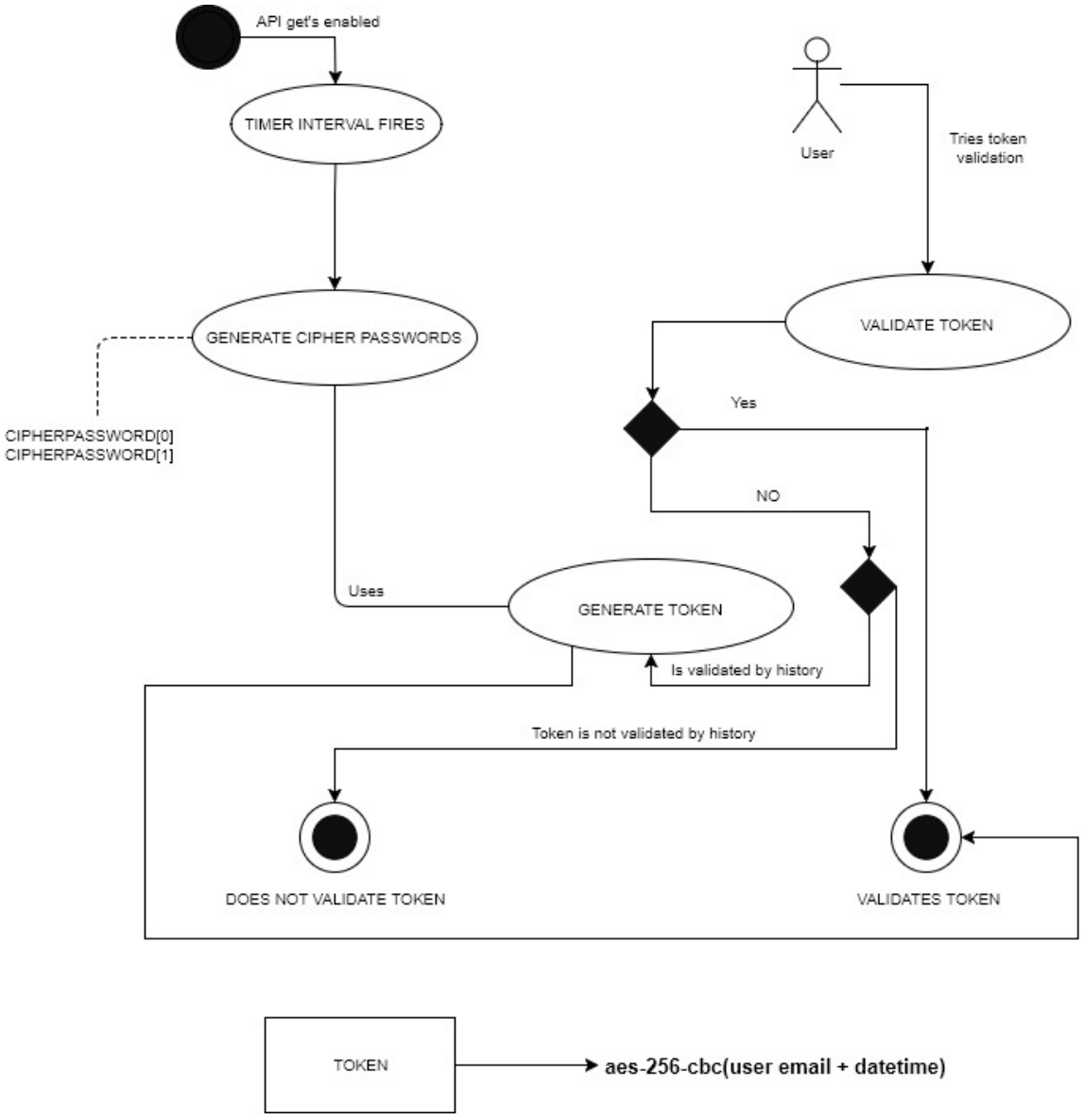

Fig. 6.6: User Validation Process

As soon as the authentication system API is online, a constant timer, responsible for generating and update an array of two sixteen character length passwords is triggered. These passwords are called cipher passwords and aim to generate user tokens that contain an hash of the user email and current token generation date.

Each token will be generated with the newest cipher password in the array and each time the timer is triggered, the oldest password is updated to the newest password and a new password is generated and saved as the newest password.

Once a request is received by the authentication system, an attempt to decipher the user token with the latest cipher password is made. If this attempt is successful, the action is allowed.

Otherwise, another decipher attempt is made with the oldest cipher password. If it is successful, a new user token is generated with the updated cipher password and the action is allowed. If the attempt fails, the action is not authorized and the user is automatically logged out.

6.3 Deploying Sprint Task

Considering the size of Sprint Task architecture, the deployment stage of the project will require multiple focus points. For example, the Authentication System and the Sprint Task API are core elements of the platform and will require frequent analysis and adjustments.

Adopting a long-term perspective, the deployment stage will occur multiple times through the continuous delivery system. For this reason, updates and deployments need to be previously scheduled, integrated and tested, in order to avoid production problems.

Aiming to prevent problems and bugs on the production version, Sprint Task will always maintain two active versions.

- Beta version
- Production version

The beta version, on the first phase, aims to handle production-like tests from referenced testing companies and test the efficiency of Sprint Task methods while they are observed and improved.

After achieving stability on Sprint Task methods and platform, a production version will be released and the existent beta version will only be responsible for handling development tests.

The following diagram aims to clarify the deployment approach of the Sprint Task project.

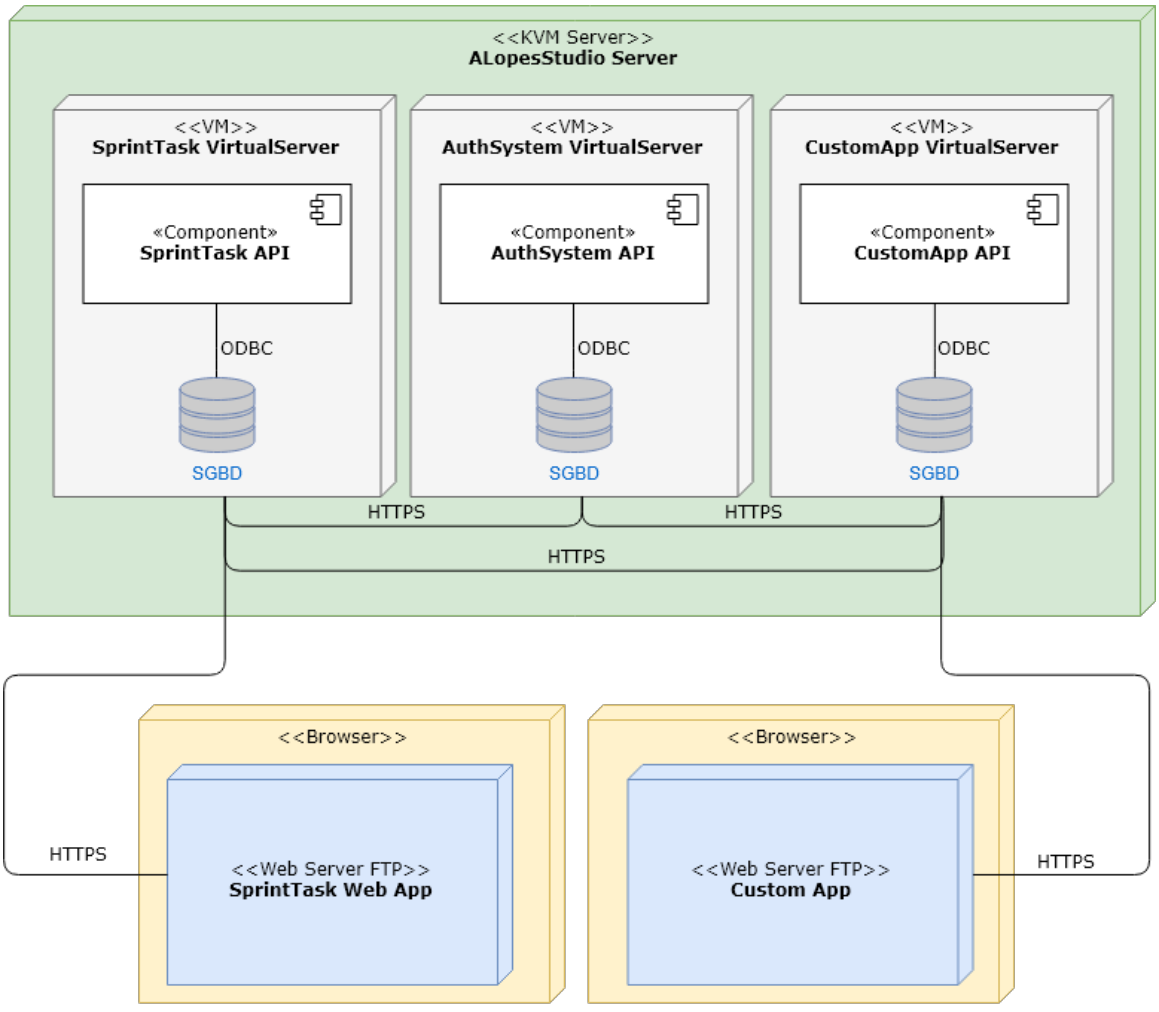

Fig. 6.7: Deployment Diagram

6.4 Approaching Mobile Users

In order to allow the full usage and experience of the platform on mobile devices, ALopes Studio aims to develop native applications based on React Native for Android and iOS. This solution will not require mobile users to access Sprint Task via browser and allow the memory management to be done differently on smartphones, improving their performance and usability. Also, adopting this approach will allow mobile users to avoid browser cache systems in the production phase.

6.5 Continuous Delivery System

In order to maintain every application updated, a continuous delivery system will be implemented on each system of the platform, upon its release. Once on production phase, it will be responsible for deploying the latest stable version when scheduled and executing frequent tests to the production version, in order to find flaws and prevent future errors.

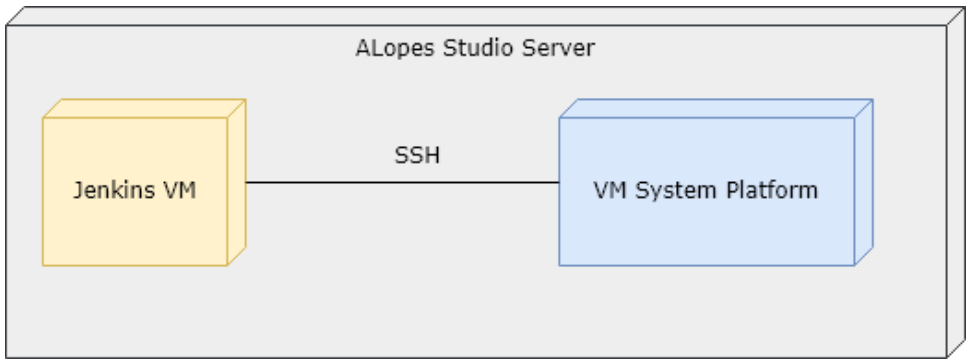

Fig. 6.8: Continuous Delivery

As a stable open-source software for continuous delivery, Jenkins reunites all the tools for managing Sprint Task platform updates to each system.

In order to update each system, Jenkins will execute customized jobs for each ALopes Studio VM Server via SSH and archive important generated artifacts of the realized tests.

As explained in the previous chapter, native applications built with React Native will be available for mobile devices, in order to improve the platform usability and efficiency. Updating a mobile application requires an upload process to the application stores, either on Android and iOS. The Apple store reviewing process is delicate and can delay the version release on iOS users. Also, mobile users are not forced to update their apps and the developers are required to maintain the support for each released version, which is expensive and time consuming.

In order to avoid unnecessary costs and possible compatibility bugs on supporting multiple versions of the Sprint Task platform, the mobile applications will automatically update themselves based on Microsoft Code Push technology.

6.6 Business Continuity Plan (Informal)

In order to support Sprint Task methods and maintain an high trust rate in the companies, the Sprint Task platform needs to be supported by a business continuity plan. Data control and recovery is crucial for every business areas because efficient companies can not rely on uncertain data.

Sprint Task relies on real-time work-flow data and operates with companies key information. Consequently, it is important to quantify the system recovery times and define backup plans.paragraph

In an initial phase, the needs for a business continuity plan are low because of the methodology improvements and the platform testing phase. However, a formal business continuity plan will be prepared before starting the production phase.

6.6.1 Beta Phase

During the beta phase, the software will be updated frequently. Companies will test Sprint Task methods in a controlled but unstable environment. The software platform will be updated whenever necessary regarding the companies feedback until the results match the predefined acceptance criteria.

Regarding the described environment, on this phase, the business continuity plan will focus on guaranteeing that companies data is not lost over platform updates. To achieve this goal, a backup of every VM System every day and every time that the platform is going to be updated.

These backups will be stored in ALopes Studio NAS System every time a backup event is triggered, either manually or due to a timed event.

Backups will be automated by shell scripts that are responsible for backing up the virtual systems in the platform and store them in the ALopes Studio NAS System and separate the outdated backups from the updated backups.

Meanwhile, the NAS mirroring system is activated and will be creating a copy of the NAS main disk every week. This allows the data-flow to be backed up frequently and secured in case of an hardware failure.

Finally, the mirrored disk will be backed up an external storage "dead storage" and stored in a safe location, outside of the company facilities.

The following diagram aims to create a visual overview of the described process.

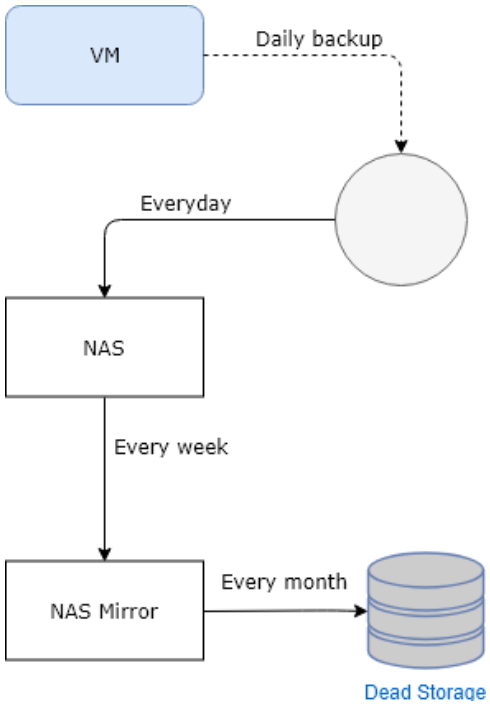

Fig. 6.9: Business Continuity Plan - Informal Scheme

6.6.2 Production Phase

A production phase environment is different from a beta phase environment.

Firstly, the platform updates are scheduled, tested and reviewed.

Secondly, ALopes Studio will develop automated software agents to handle platform recovery actions. These agents will handle server-side malfunctions by reacting to failures on pre-created testing scripts, that are executed frequently. For example, triggering backup events, alerting the responsible personal and realizing redeployment actions can be reactions to the testing scripts.

During the production phase, the backup plan will be maintained and system recovery simulations will be executed, in order to reduce the amount of time that the Sprint Task platform is unavailable.

6.7 Positioning Sprint Task With GDPR

Sprint Task is based on companies work-flow data and, for that reason, it is responsible for maintaining their data confidential and secured. The data stored in Sprint Task is able to reveal important confidential work procedures if exposed. Due to the importance of the data, maintaining it secured, confidential and backed up is a critical concern of the Sprint Task team.

The new GDPR (general data protection regulation) is responsible for guaranteeing the data confidentiality and safety. Designed for the software industry, companies are not allowed to treat users data freely anymore. The GDPR has established a set of rules for storing and transacting data. Moreover, the user rights have also been published by this community regulation.

Regarding the user rights over their data, it became important to position Sprint Task according to the new community regulation. Sprint Task is a proprietary software that will not allow an online registration or subscription in an early phase. Companies are required to contact ALopes Studio to subscribe a Sprint Task license and sign a contract that aims to specify which points of the GDPR are not totally fulfilled.

Due to the mandatory software relationships and dependencies between users, teams, projects and tasks Sprint Task assumes a strict position regarding core data, as spefied for the following points:

- Users data will not be used for marketing purposes, however, the Sprint Task platform may react to situations with invasive approaches, like push-notifications and automated procedures that may alter data for a sustainable usage of Sprint Task methods.
- In order to allow users to remove their acknowledgement of Sprint Task data treatment procedures, the full company contract needs to be revoked, to prevent the misapplication of data.
- Sprint Task allows data portability under customized procedures. Due to the synchronized applications with the main platform, these procedures are different for each company and may require different times and costs.
- Users are not allowed to edit or delete any data, unless it is predicted or allowed by Sprint Task methods.
- Users data will be conserved and utilized over time and will not be deleted until the contract is revoked.

Evaluation

7.1 Testing Sprint Task Methods

At the of this dissertation, Sprint Task was not ready for a beta phase. The platform still requires improvements that are crucial for the software sustainability.

As explained before, a group of companies is already defined to integrate Sprint Task beta phase. The aim of the beta phase is to gather real work-flow data and prove the efficiency of Sprint Task methods.

To achieve these goals, a detailed analysis of companies work-flow procedures will take place before the integration of the Sprint Task methods and platform. Aiming to achieve quantitative work-flow analysis regarding discipline and agility, a set of forms will be answered several times regarding its context. These forms aim to analyze the procedures efficiency, the collaborators efficiency and the managers planning skills.

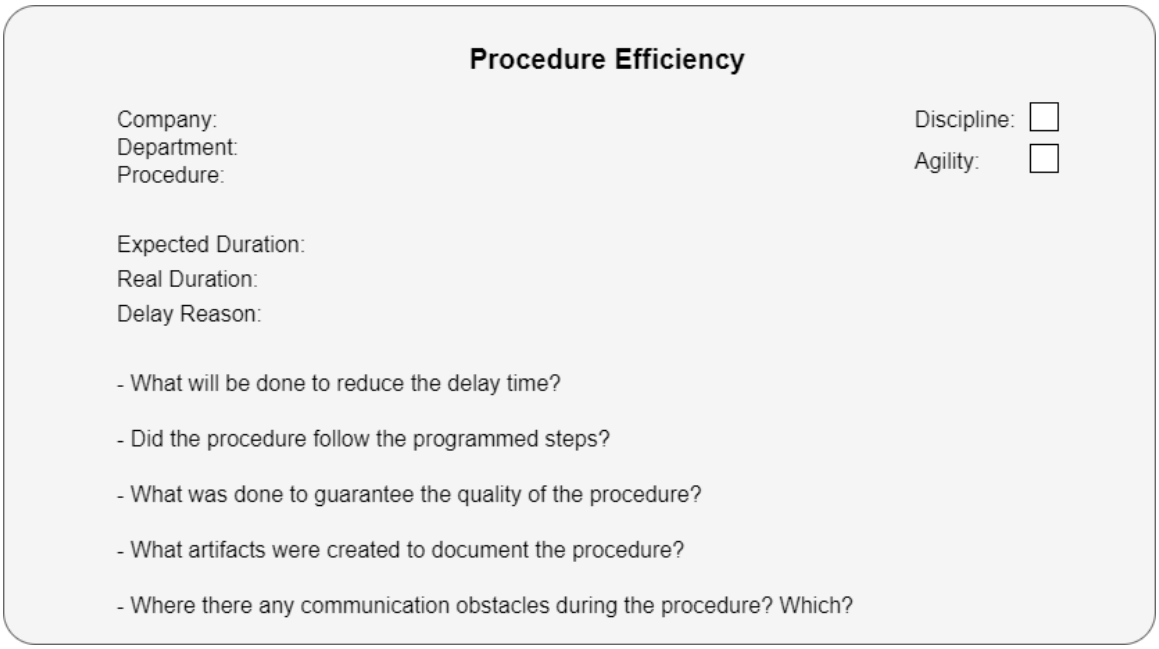

Fig. 7.1: Procedure Efficiency Form

To evaluate the procedures efficiency, it is important to compare the estimated procedure quality standards and its estimated duration with the real values. Repeating this analysis multiple times with the same procedure to generate enough sustainable data will allow the procedure analysis to be sustainable and reliable.

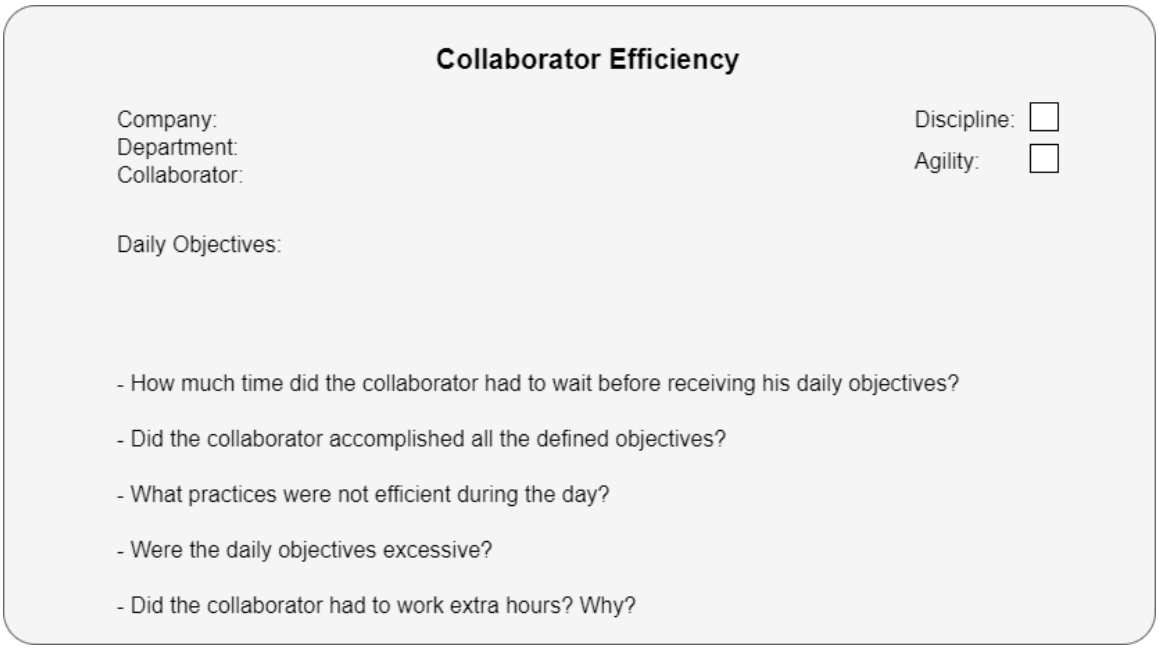

Fig. 7.2: Collaborator Efficiency Form

Another key factor to take in consideration during the work-flow efficiency analysis is the collaborator's efficiency. Usually, collaborators loose considerable amounts of time waiting for hierarchic orders due to the nonexistence of autonomy.

Also, their motivation is an important key factor for the success of their daily objectives. The motivation levels can be reduced for the excessive work tasks and continuously unsuccessful objective accomplishment. For that reason, it is important to gather information about the amount of planned work tasks and realized extra hours by collaborators.

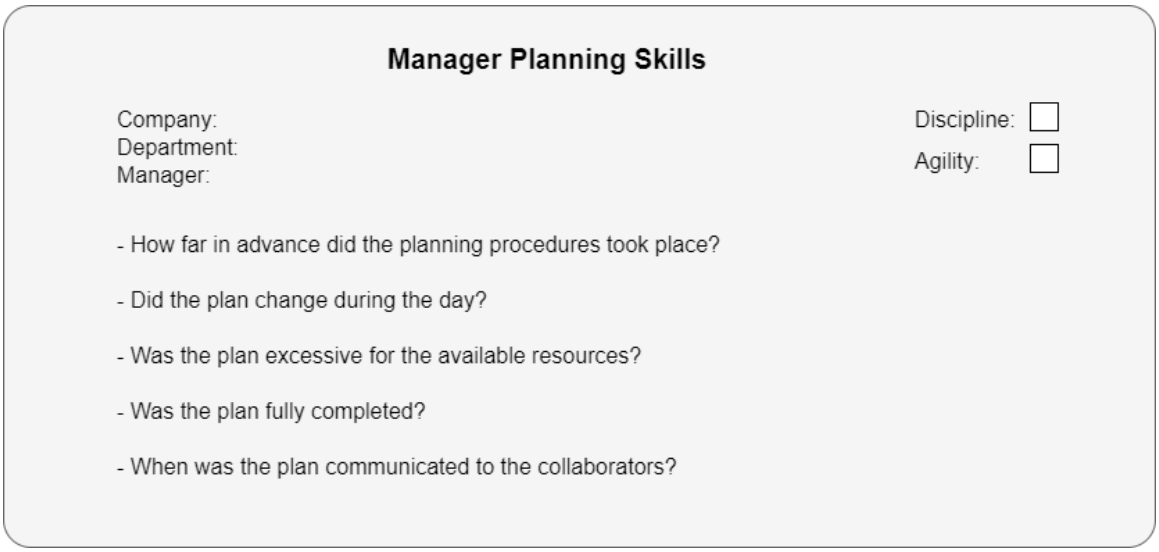

Fig. 7.3: Manager Planning Efficiency Form

The last form aims to evaluate the managers planning skills, in order to understand how timed planning decisions are made, how appropriate and stable they are and how successful the manager plans are.

In order to gather an average evaluation of the companies performance efficiency, a time interval will be set to analyze the planning practices, collaborators performance and procedures efficiency. The previous forms will be answered several times in order to gather enough data before implementing Sprint Task, which depends on the amount of selected analyzable procedures, collaborators and planning events. It is important to note that each form will be repeated for a minimum of five times.

After the data gathering phase, the Sprint Task implementation takes place. During this time, an effort to familiarize managers and collaborators with Sprint Task methods and platform, is crucial to allow companies to take fully advantage of the methods. It is important to reference that this implementation will be guided by someone in Sprint Task team.

Once the company is prepared to deal with Sprint Task on its own, the responsible guide will encourage the company to work freely with the platform during one month. Only the, the data gathering phase will be repeated and compared to the previous gathered data. In the next figures, it is exposed an optimal result of this comparison for each different form.

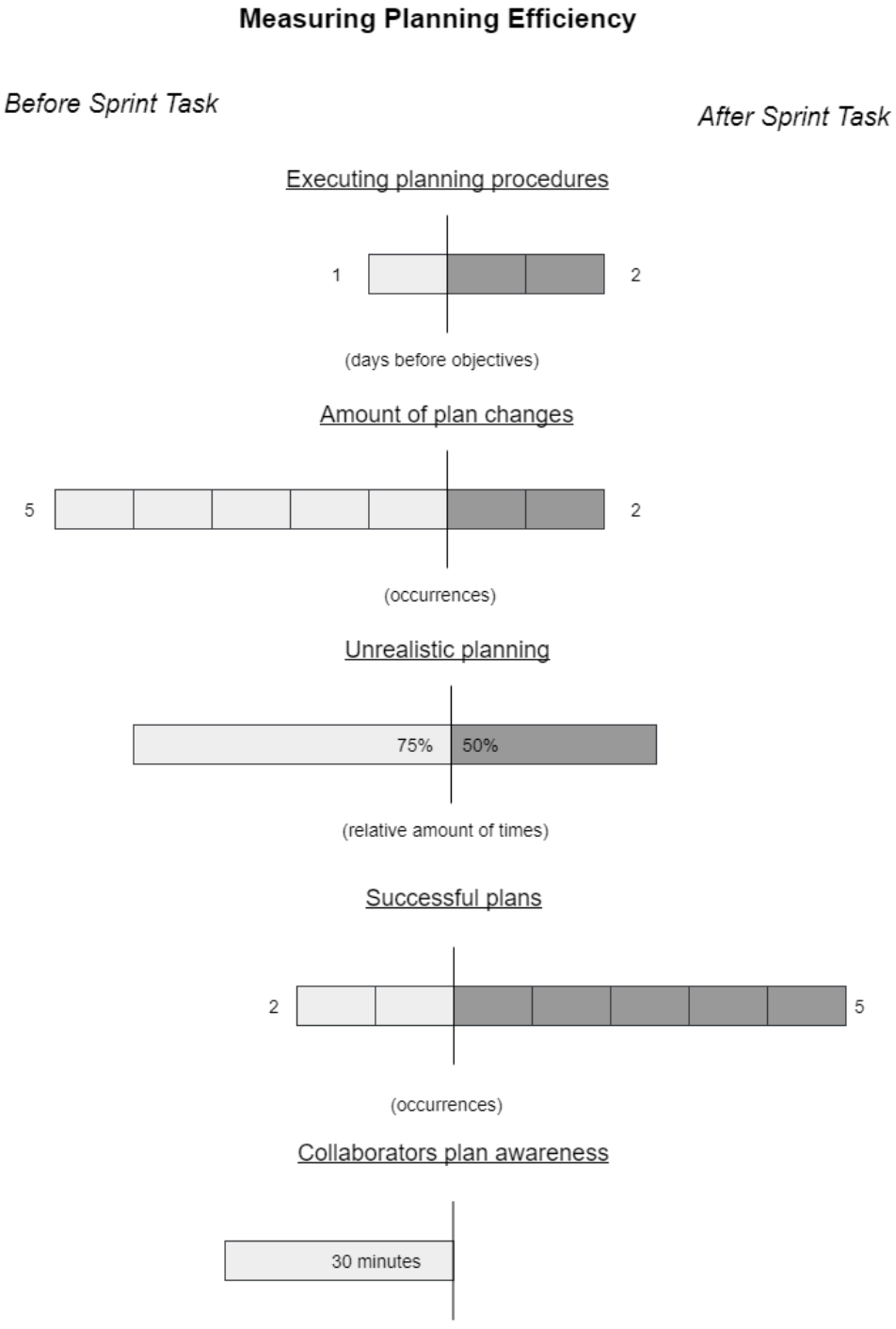

Fig. 7.4: Measuring Plan Efficiency

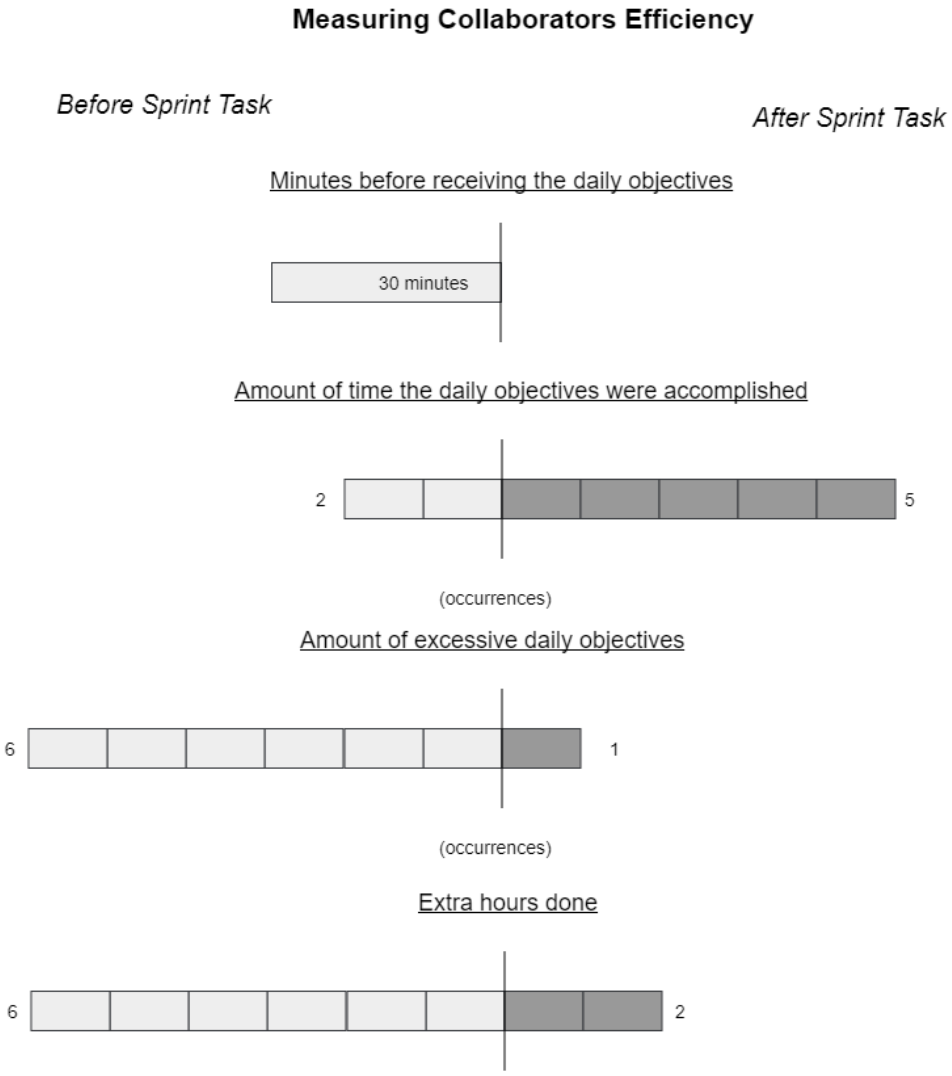

Fig. 7.5: Measuring Plan Efficiency

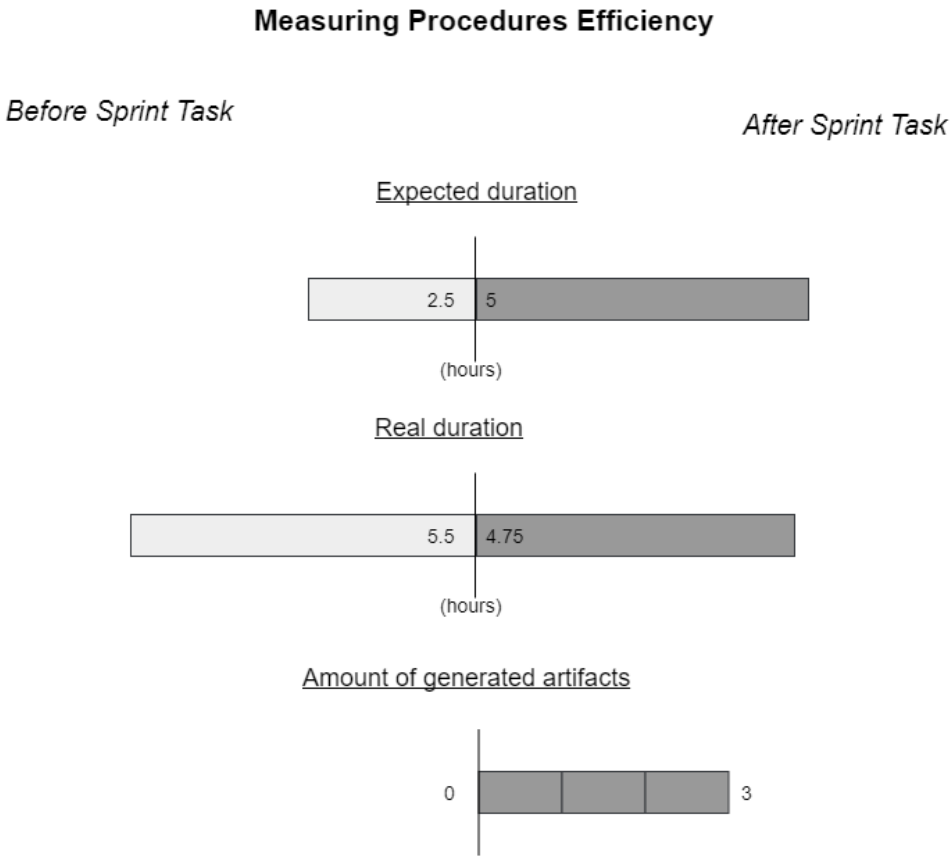

Fig. 7.6: Measuring Procedures Efficiency

According to the previously presented comparison, it is possible to extract Sprint Task goals on companies work-flow impact.

- Improve planning skills
- Anticipate planning tasks
- Reduce the amount of unpredictable events
- Increase plans realism
- Improve collaborators plan awareness and reduce their waiting time on daily goals
- Reduce the amount and impact of inefficient practices
- Reduce the amount of extra hours
- Increase the generation of standard organized artifacts that document procedures and practices

Each referenced goal is a key factor for a sustainable work-flow inside a company. It is important to maintain stable and acceptable levels of success in each point, otherwise, the work-flow may become unsustainable and expensive. Sprint Task methods aim to improve those goals intrinsically, prevent and predict future problems.

7.2 Hypotheses

The present dissertation intends to sustain and proving the following hypotheses:

- Sprint Task methods improves general work-flow environments of the common industry
- Sprint Task platform is versatile and adaptable to different business areas of the common industry

In order to prove the positive impacts of Sprint Task the common industry work-flows, it is necessary to prove that Sprint Task goals are accomplished.

Moreover, to accept the first hypotheses, it was defined by ALopes Studio that Sprint Task may accomplish at least six out of the nine defined goals, on average, considering the first beta phase comparisons.

Regarding the second hypotheses, the defined criteria demands Sprint Task comparison results to achieve, at least, four out of the nine defined goals on each beta phase analysis.

Conclusions

At the end of this dissertation, the project was not finalized. In fact, it was not possible to start the project beta phase in the selected companies due to the project low maturity levels. Although the main application is ready to integrate the beta phase, the marketplace is still not capable of adding value to the costumers, which results in the low impact of Sprint Task in general work-flows.

The beta phase will involve all the all the analyzed companies and is set to start as soon as the marketplace can provide the key-tools for each work-flow.

On the other side, ALopes Studio believes that Sprint Task will cause noticeable improvements on common industry work-flows. Also, it is expected for companies to experience a change in their work-flows orientation. Because Sprint Task expects to transform production lines into people oriented work-flows by taking in consideration the work process stakeholders perspective.

Finally, despite in a pre-evaluation phase, Sprint Task has already highlighted the main common problems of common industry work processes and already provides a conceptual solution based on the impact of agile methods on the software industry.

References

[1] D.J. Anderson and A. Carmichael. *Essential Kanban Condensed* (2016).

[2] J.D. Arenas. *Effective practices in the Front-End of Innovation*. `https://www.linkedin.com/pulse/effective-practices-front-end-innovation-jorge-d%C3%ADaz-arenas/`, 2016. Accessed: 2018-02-05.

[3] K. Beck, M. Beedle, A. van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R.C. Martin, S. Mellor, K. Schwaber, J. Sutherland and D. Thomas. *Manifesto for Agile Software Development*. `http://agilemanifesto.org/`, 2001. Accessed: 2018-02-08.

[4] P. Belliveau, A. Griffin and S. Somermeyer: *The PDMA Toolbook for new product development*, 2002.

[5] B. Boehm and R. Turner, (2004).

[6] D. Fernandez Martin. *easyAHP*. `http://www.easyahp.com/`, 2016. Accessed: 2018-02-06.

[7] J. Highsmith and A. Cockburn, (2001).

[8] E. Ko and H.A. Burhan, (2015).

[9] D. Lowe. *Littles Law*. `http://scrumandkanban.co.uk/littles-law/`, 2014. Accessed: 2018-02-04.

[10] J.R. Pekka Abrahamsson, Outi Salo and J. Warsta. *Agile Software Development Methods: Review and Analysis*, 2002.

[11] K. Schwaber. *Nexus Guide* (2018).

[12] K. Schwaber and J. Sutherland. *The Scrum Guide* (2017).

Acknowledgements

The development of the Sprint Task platform would not be possible without the support of ALopes Studio and its departments. Getting involved in the industry department and being able to experience the work-flow in real-life was one of the most motivating and important factors for the concept development. It is important to highlight all the support from Mr. Lopes during all the project, whom always provided its own vision and perspective, but always gave me all the freedom to pursue and adopt my own perspectives and technical approaches.

In addition, I would also like to thank Fabsgarage and FAP for participating in the project development, and cooperating with ALopes Studio during the study of the common industry work-flows problems.

Also, I would like to thank Dr. Nuno Escudeiro and Dr. Mafalda Ferreira for all the guidance, availability and patience with all my questions during the project.

Last but not least, I would like to thank my parents and my girlfriend Ana Nogueira for all the love and support.

Attachments

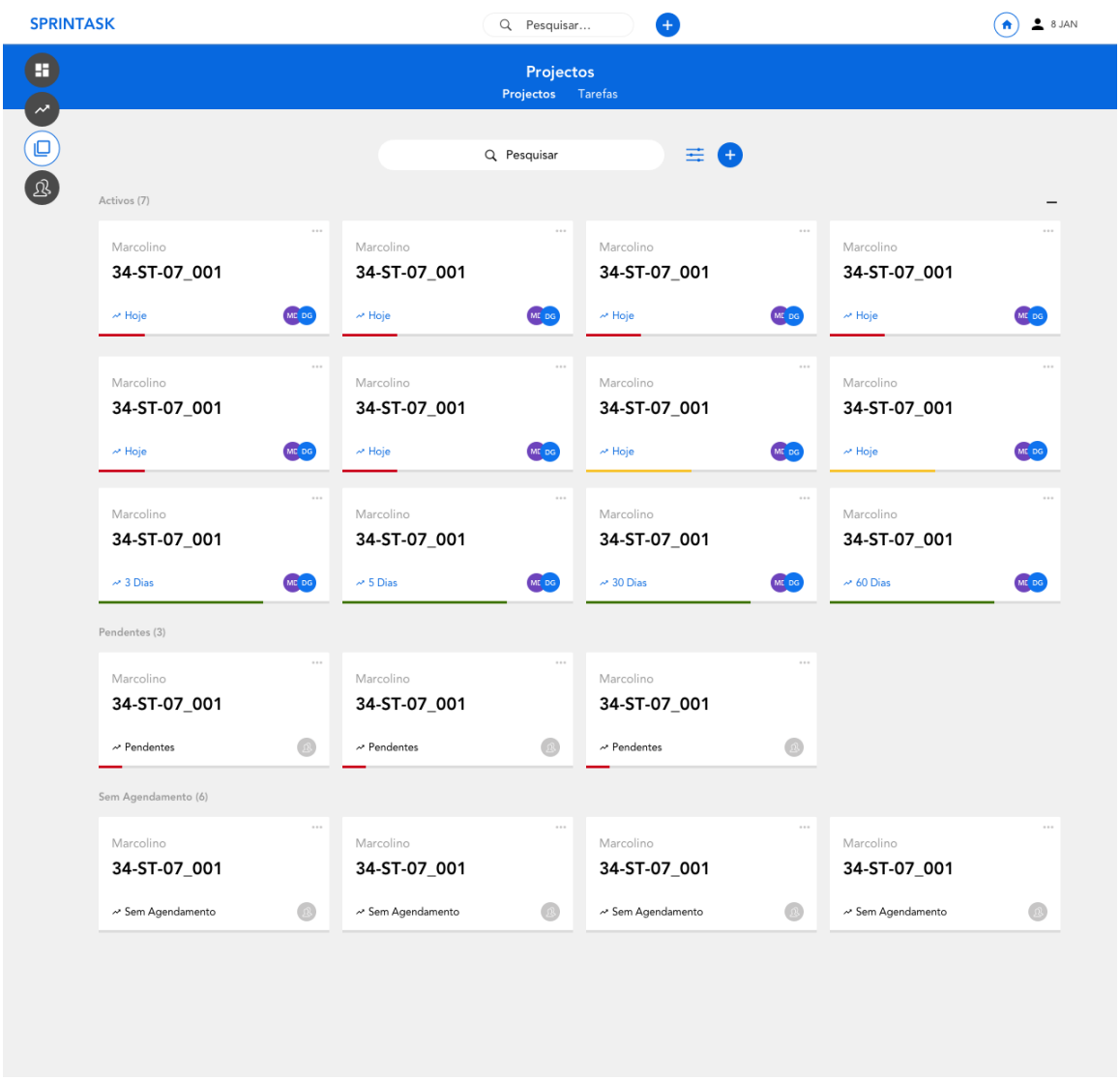

Fig. 8.1: Projects page

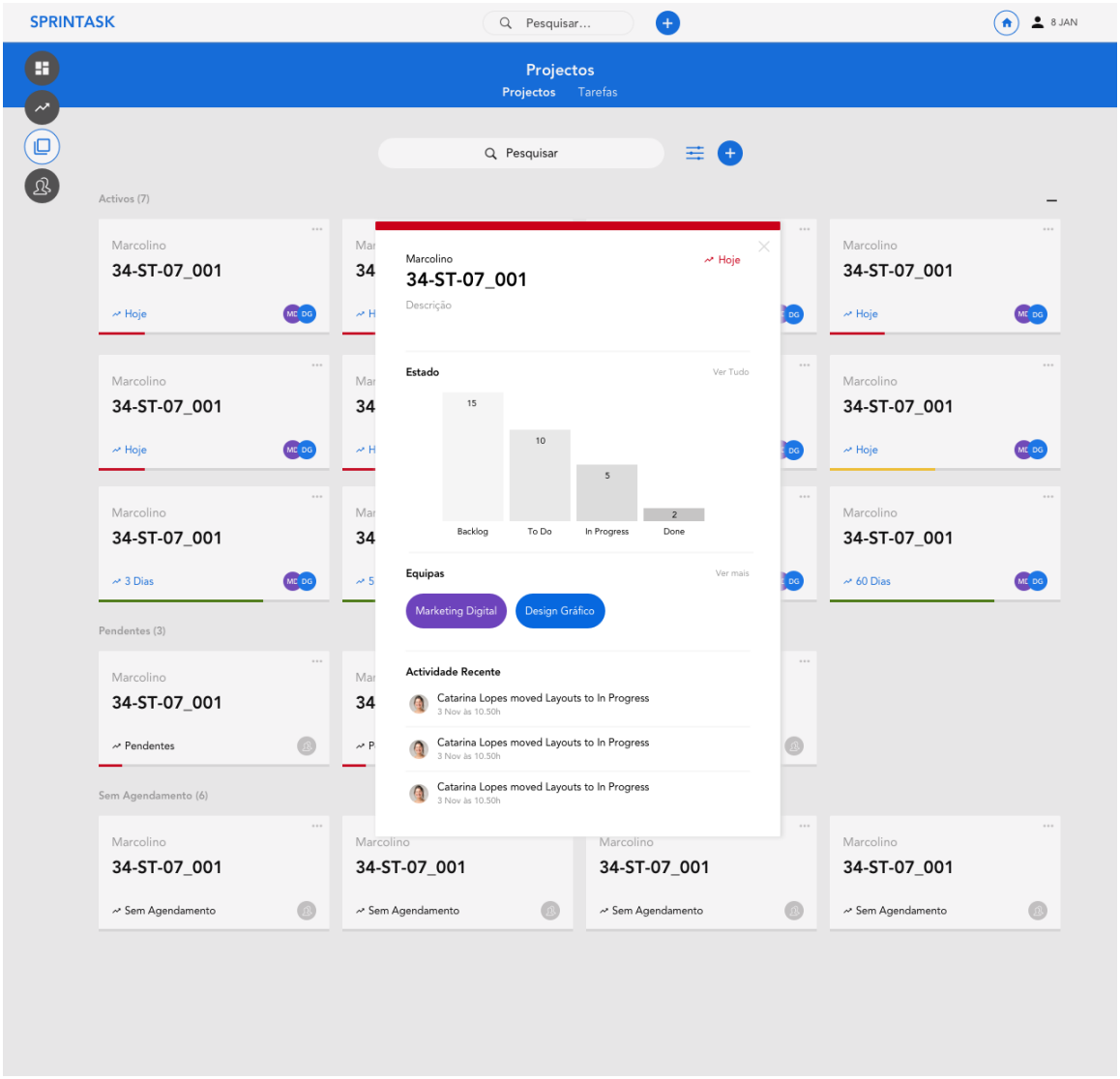

Fig. 8.2: Project quick view

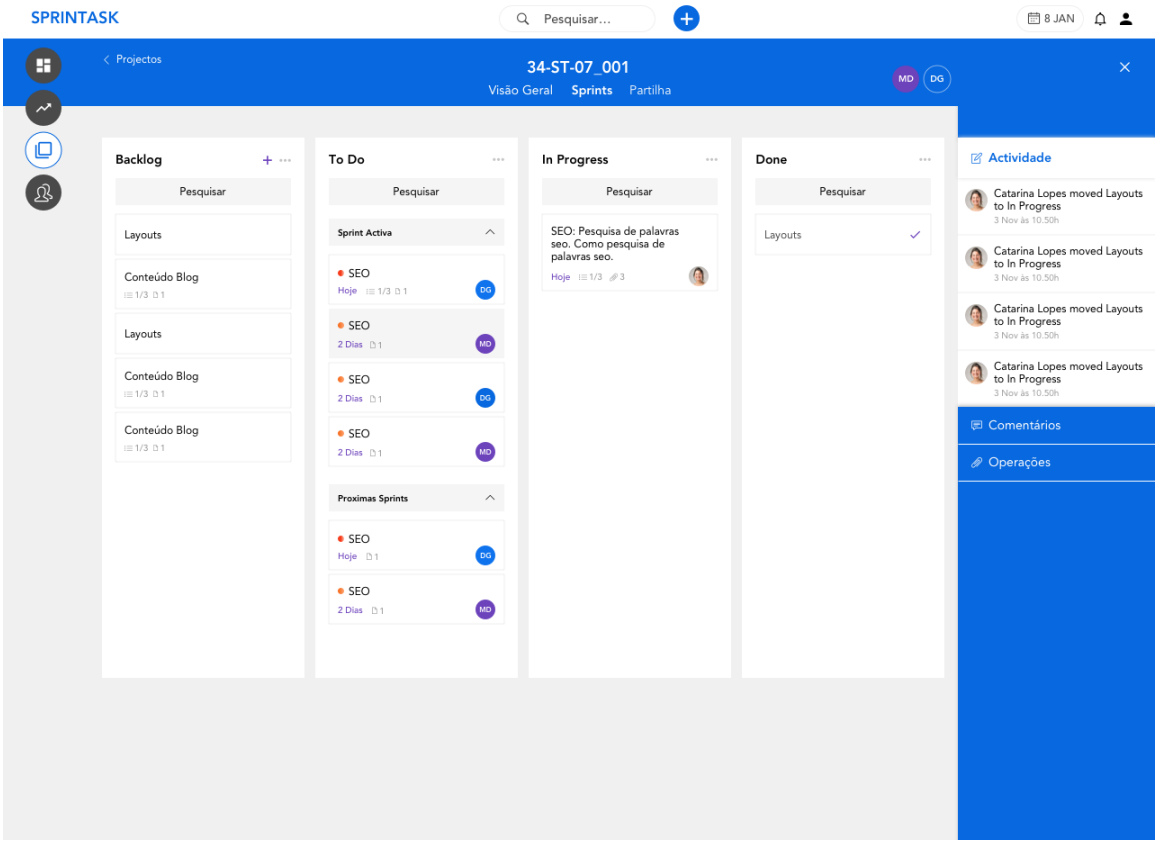

Fig. 8.3: Project sprint

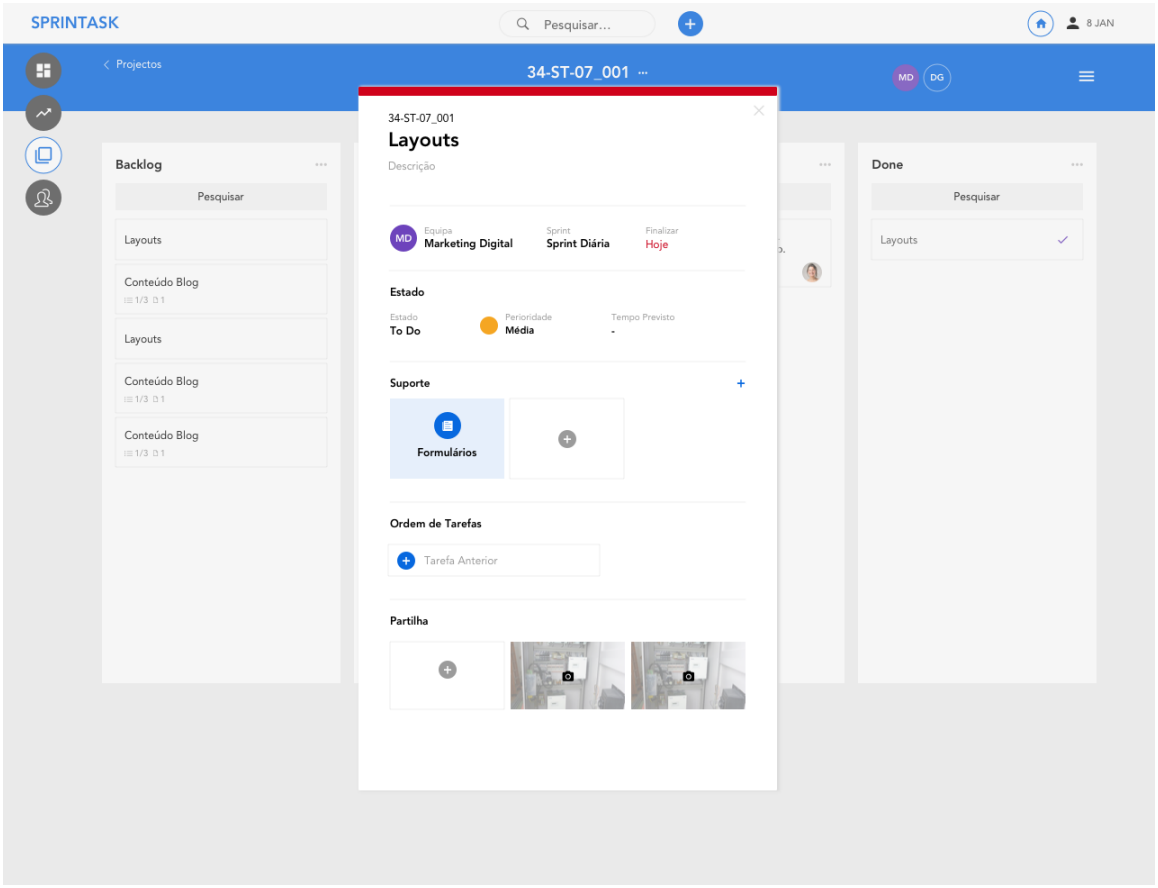

Fig. 8.4: Task overview

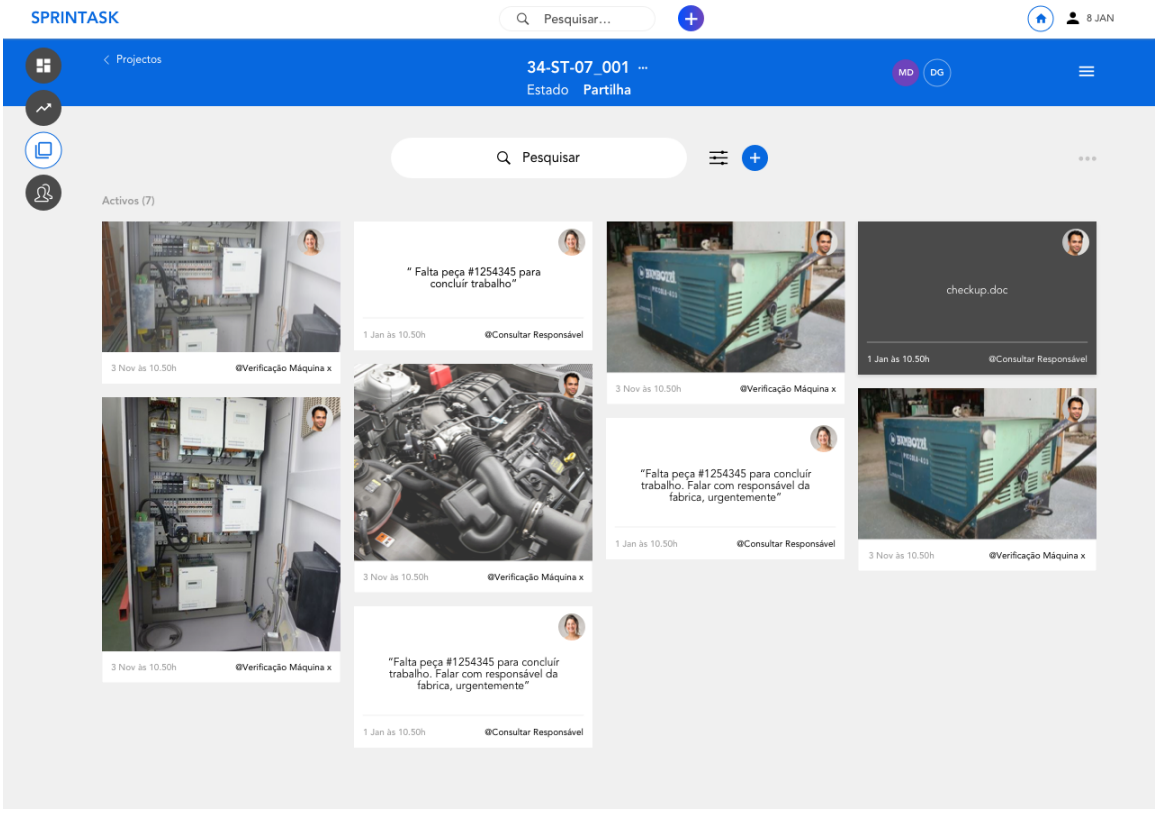

Fig. 8.5: Project attachments

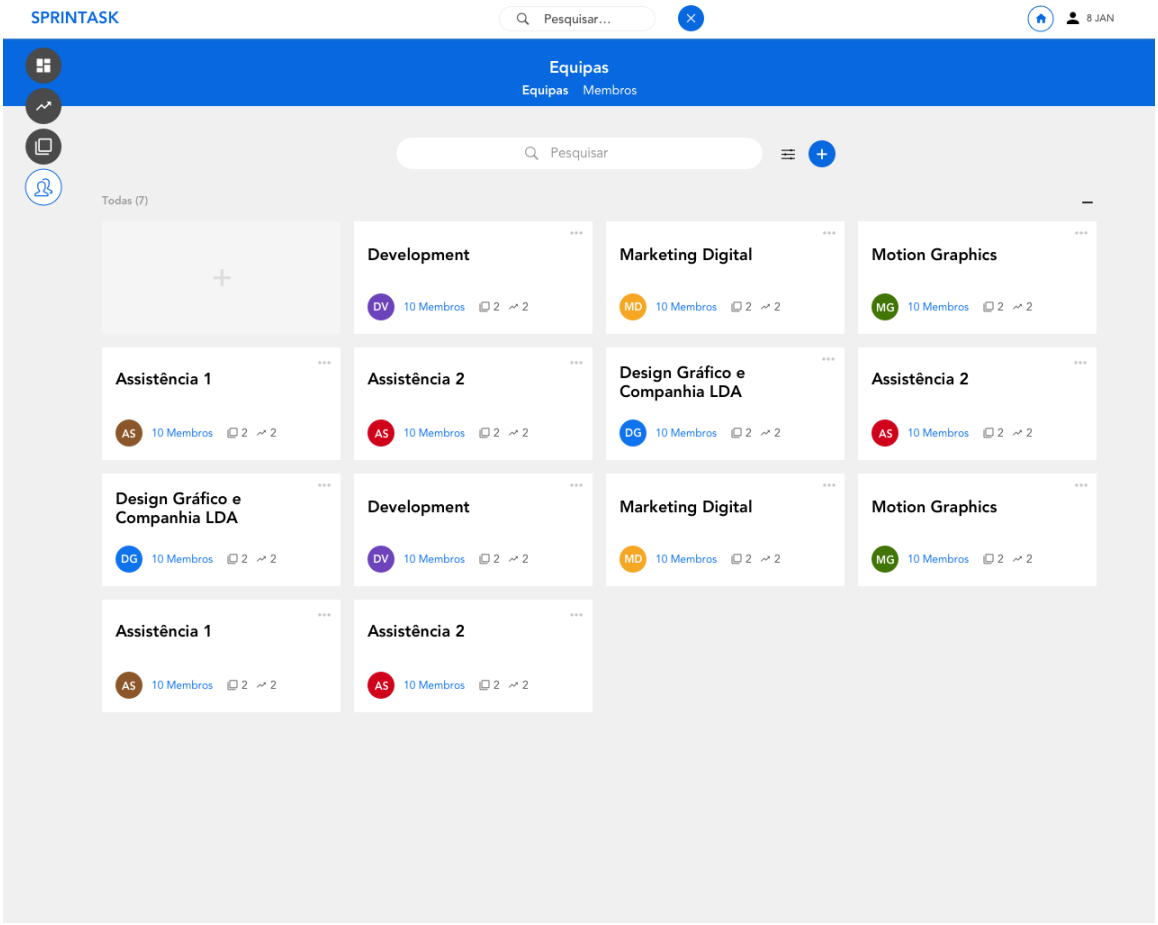

Fig. 8.6: Teams page

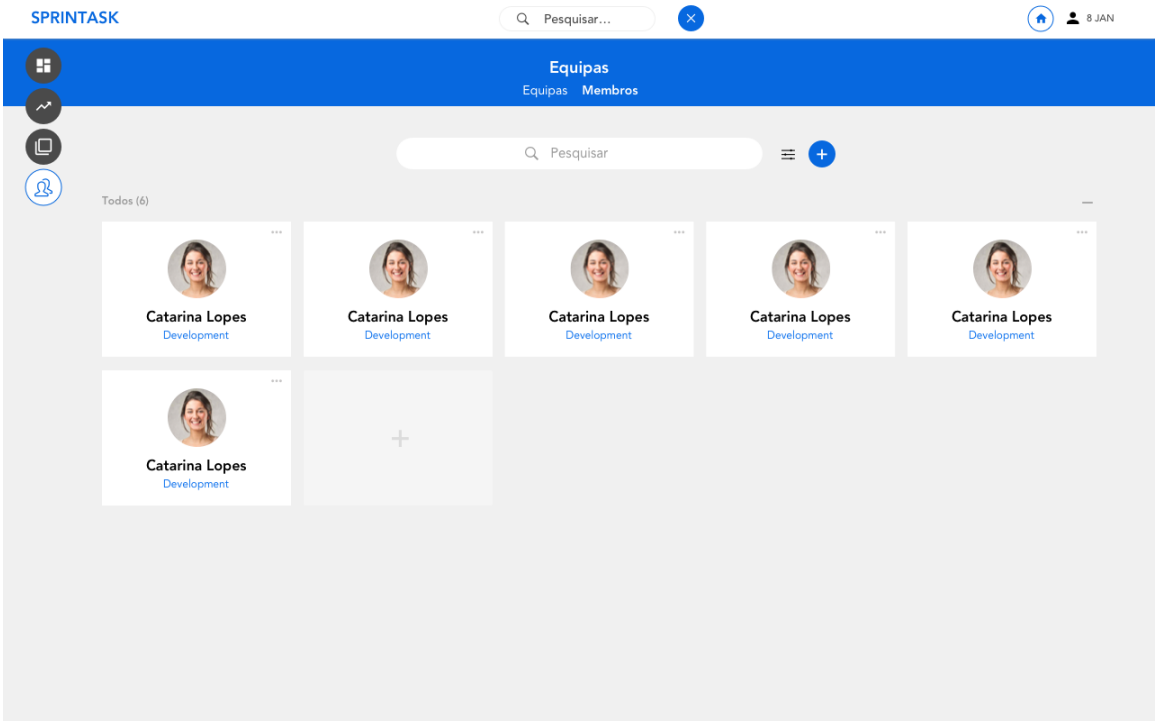

Fig. 8.7: Team overview

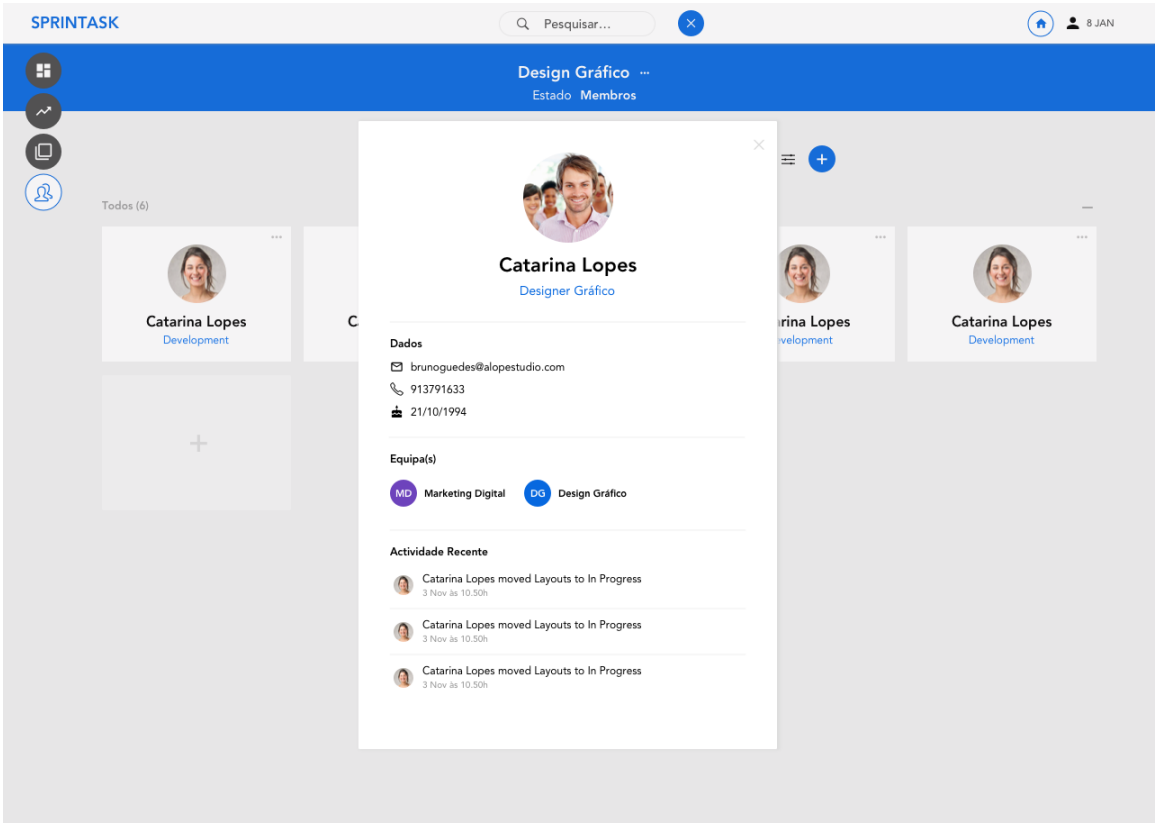

Fig. 8.8: Collaborator overview

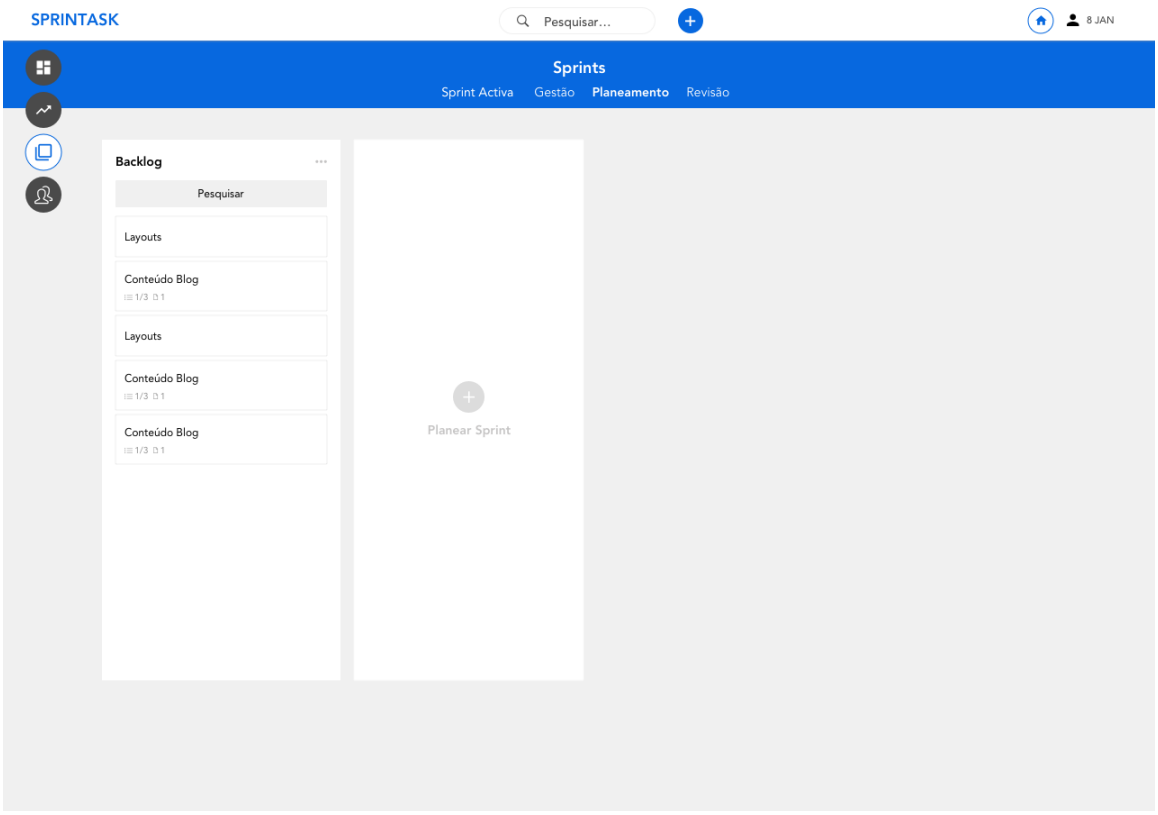

Fig. 8.9: Sprint planning - 1

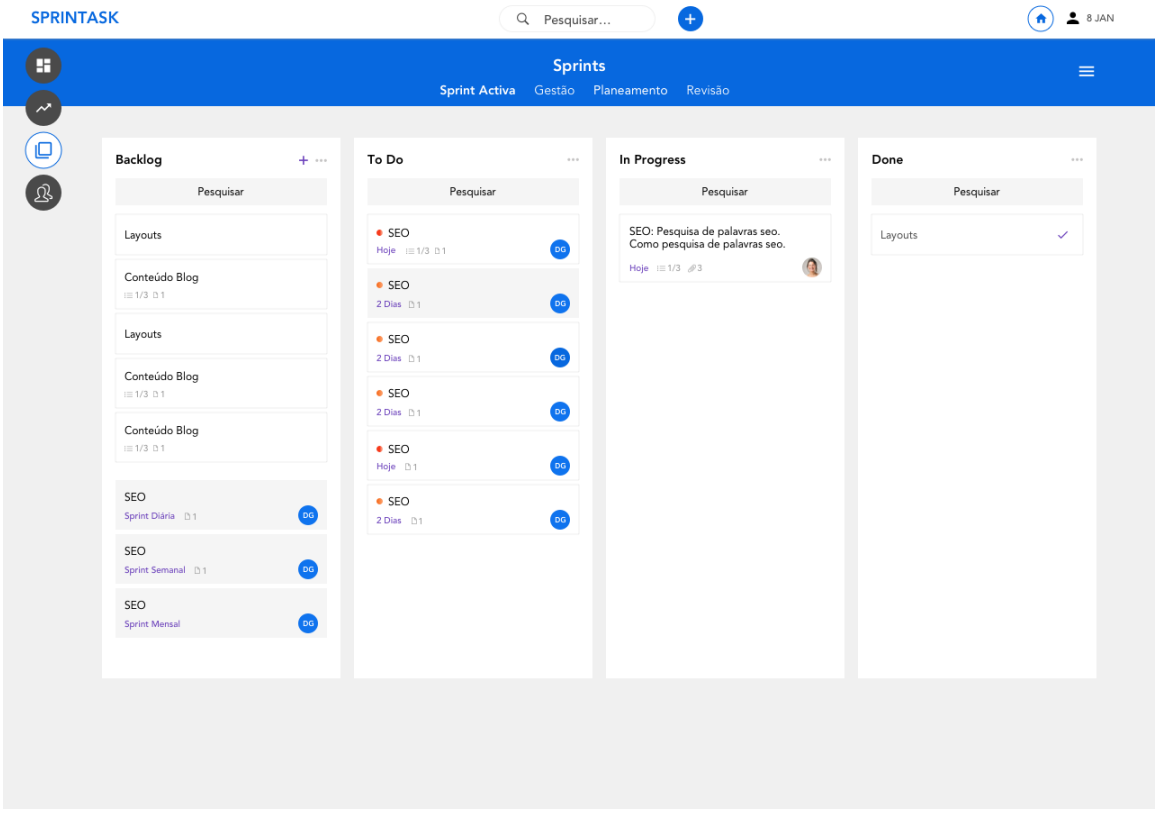

Fig. 8.10: Sprint planning - 1

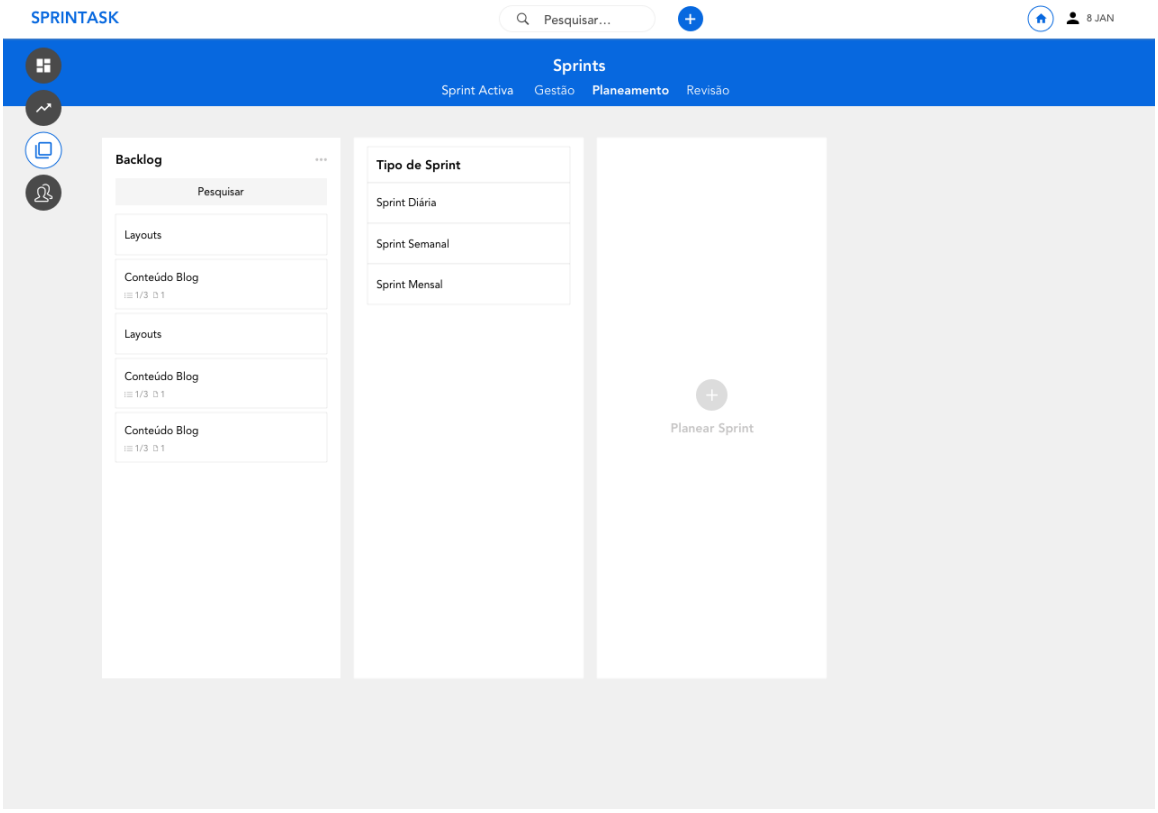

Fig. 8.11: Sprint planning - 2

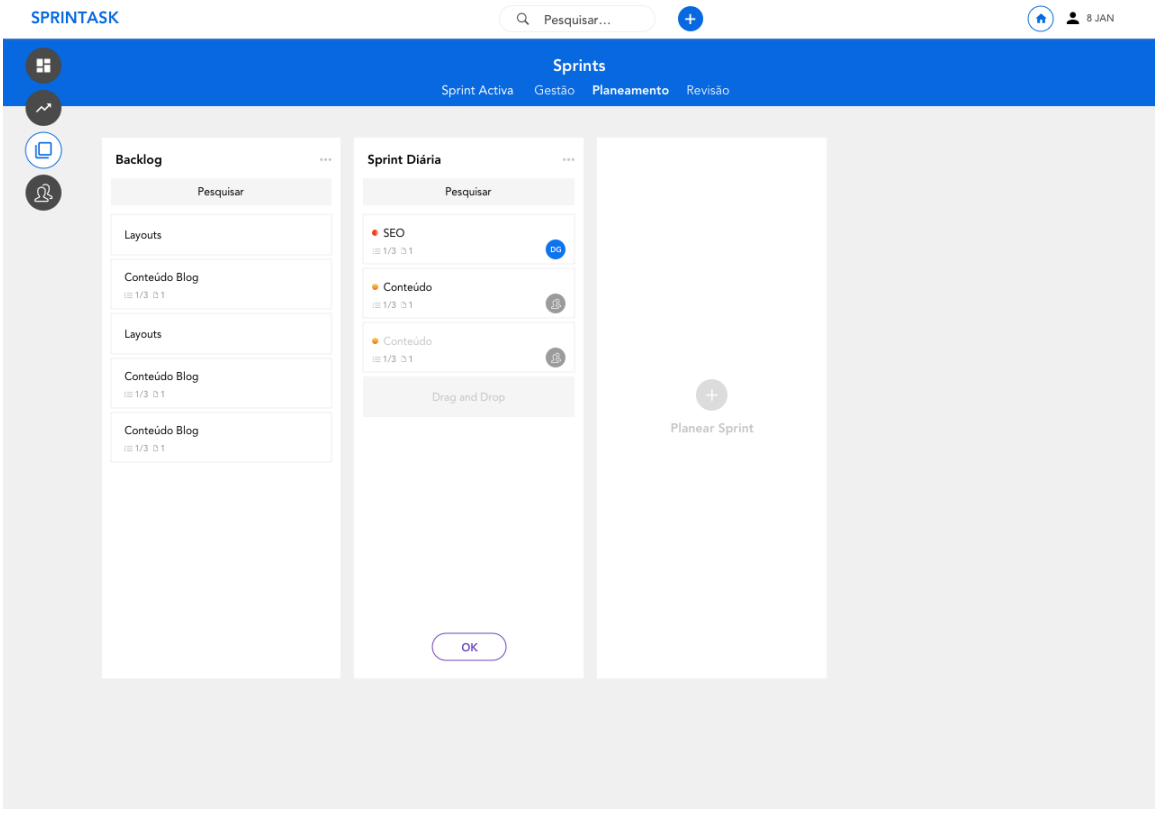

Fig. 8.12: Sprint planning - 3

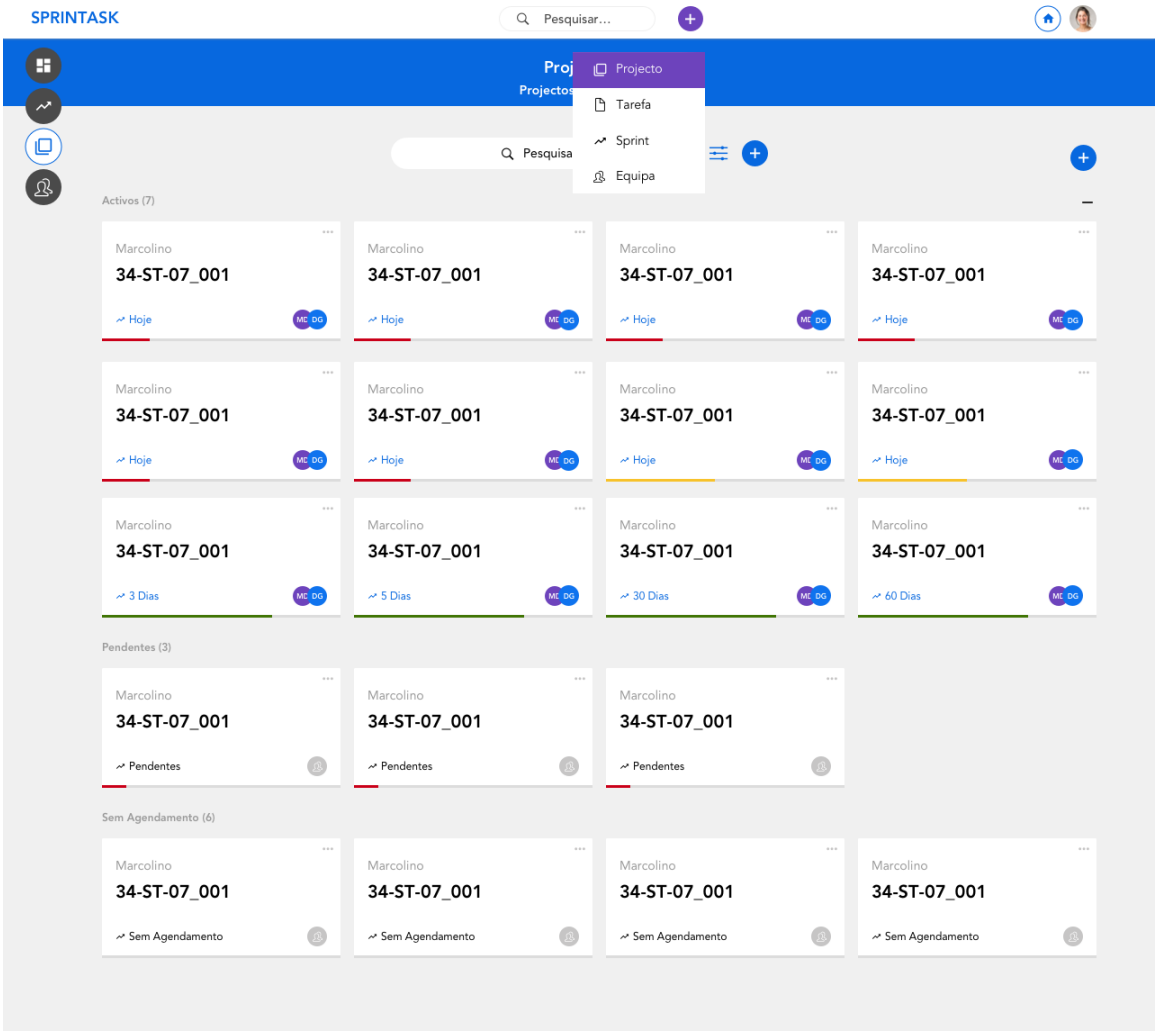

Fig. 8.13: Plus button overview

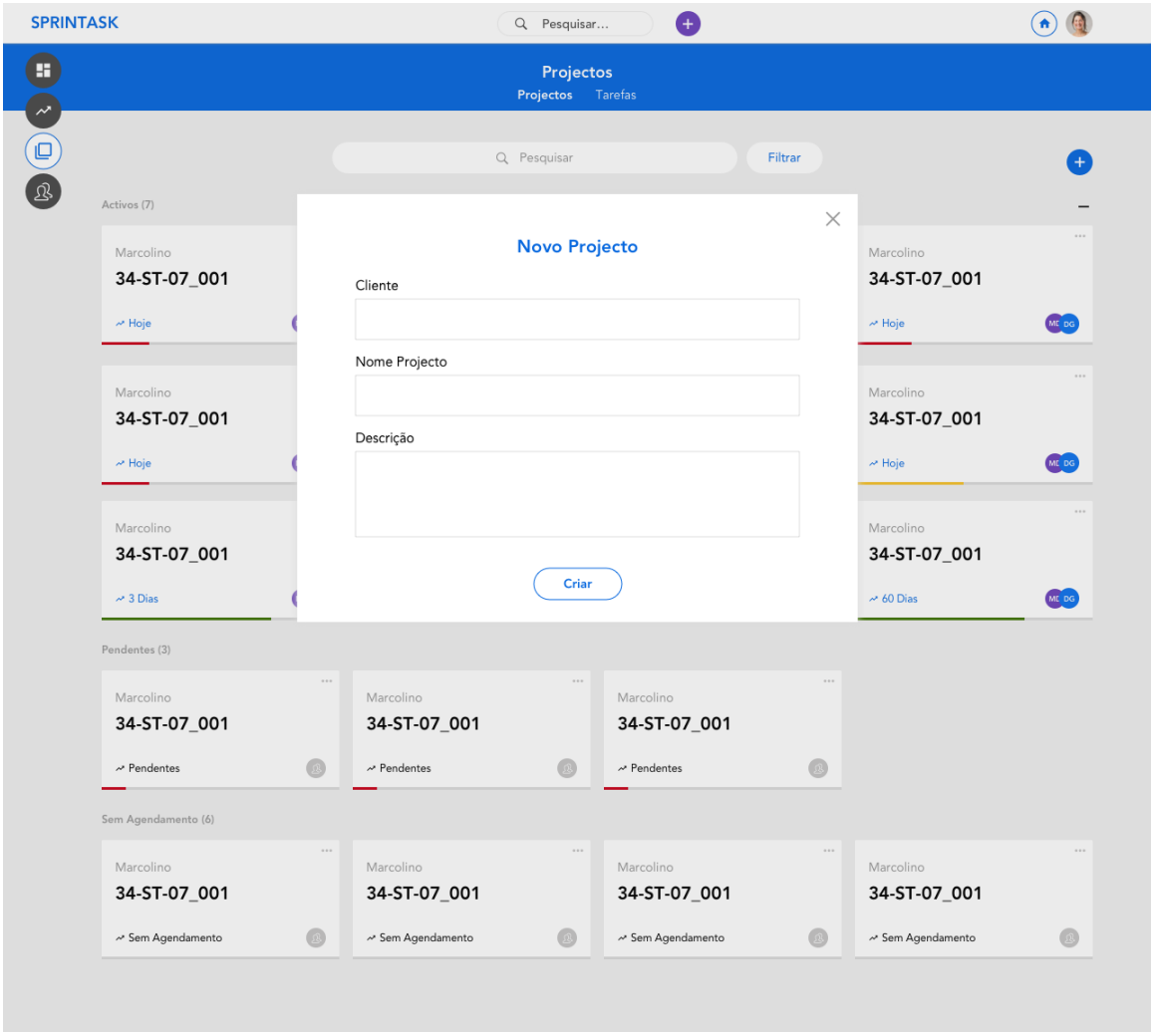

Fig. 8.14: Create project form

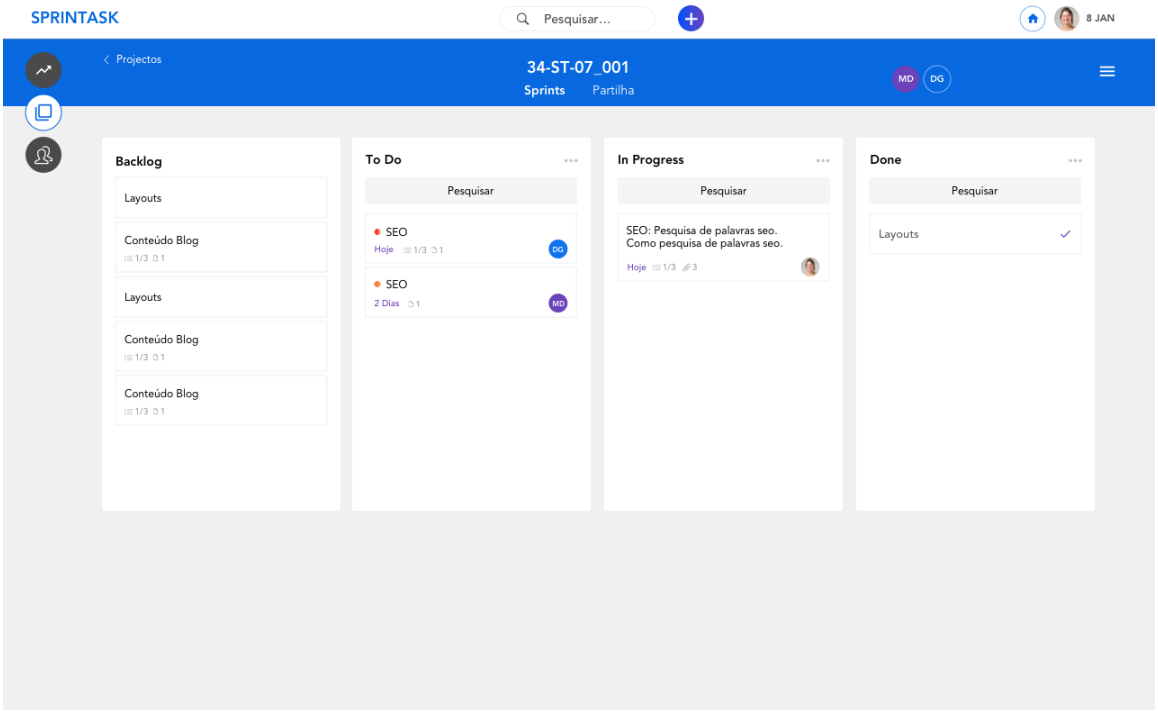

Fig. 8.15: Project Sprint

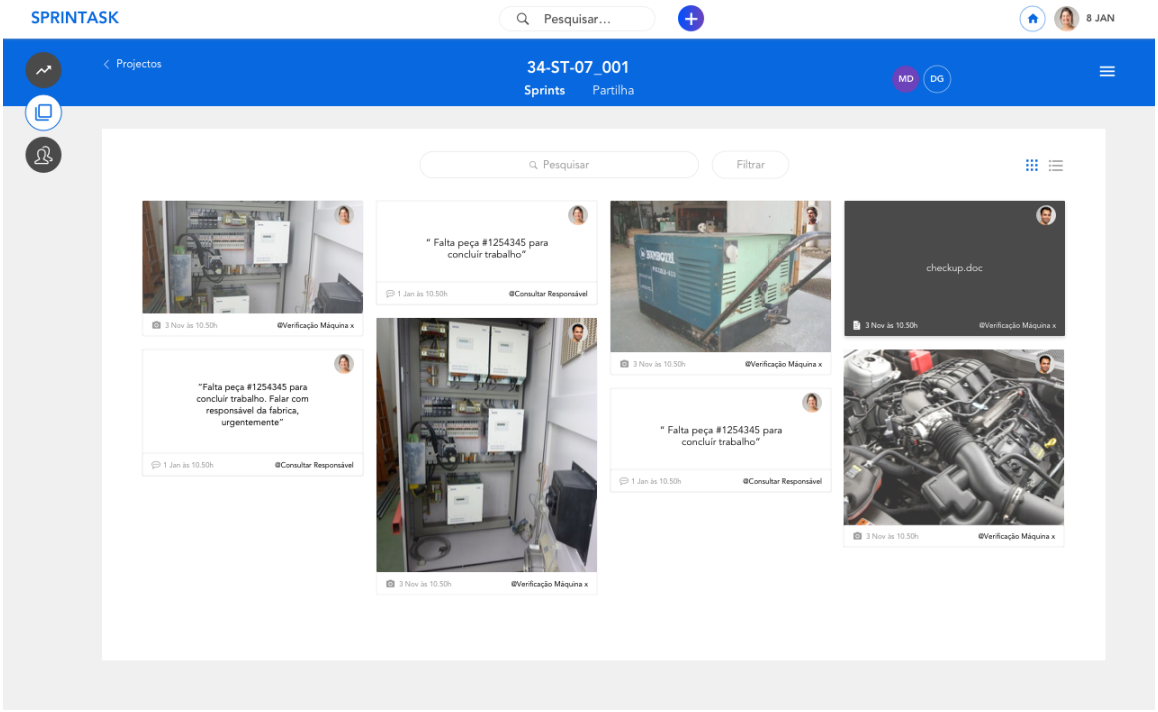

Fig. 8.16: Project attachments

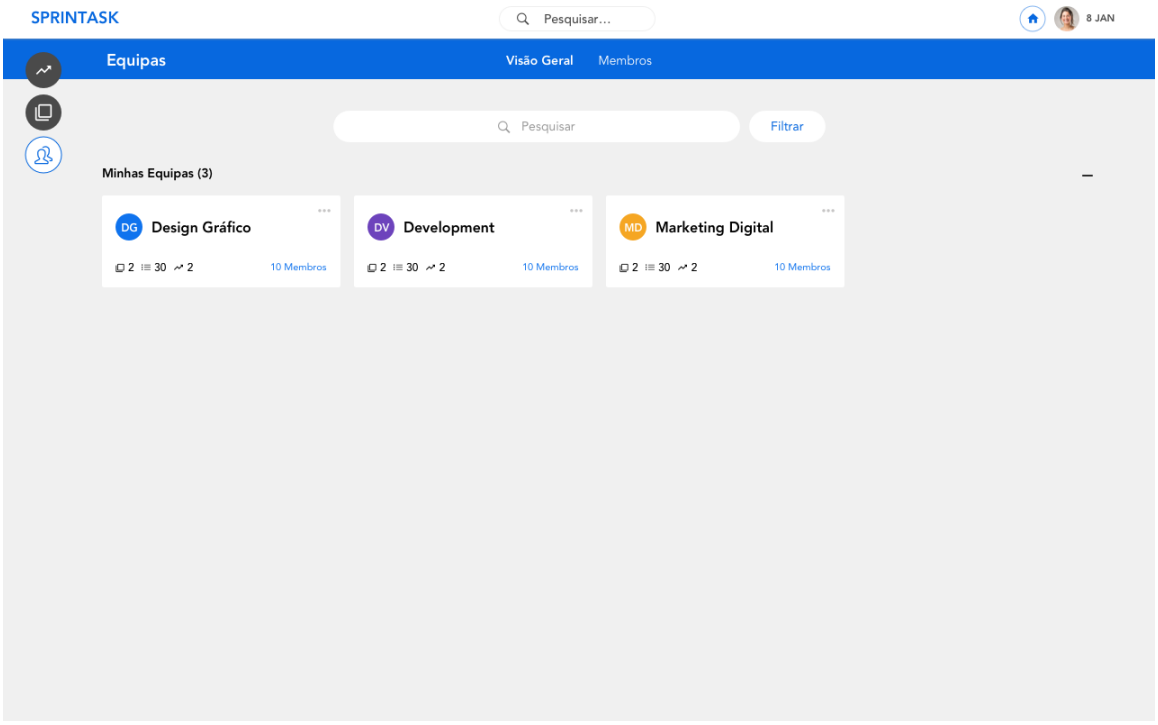

Fig. 8.17: Teams overview

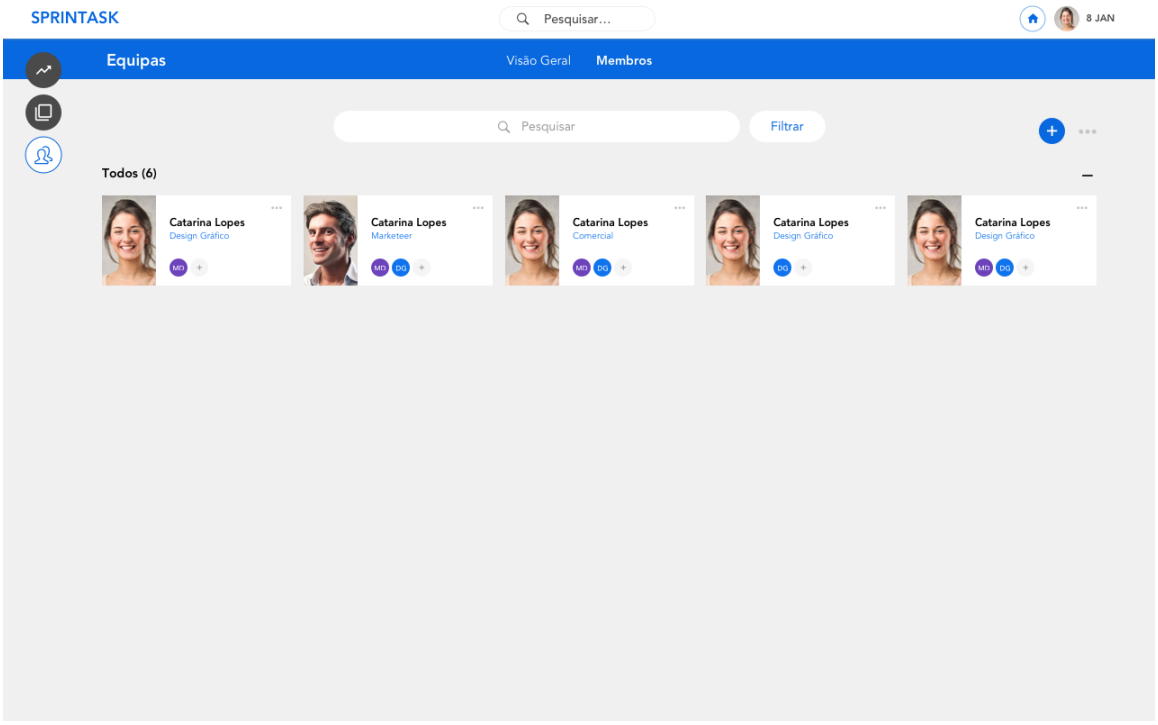

Fig. 8.18: Team view

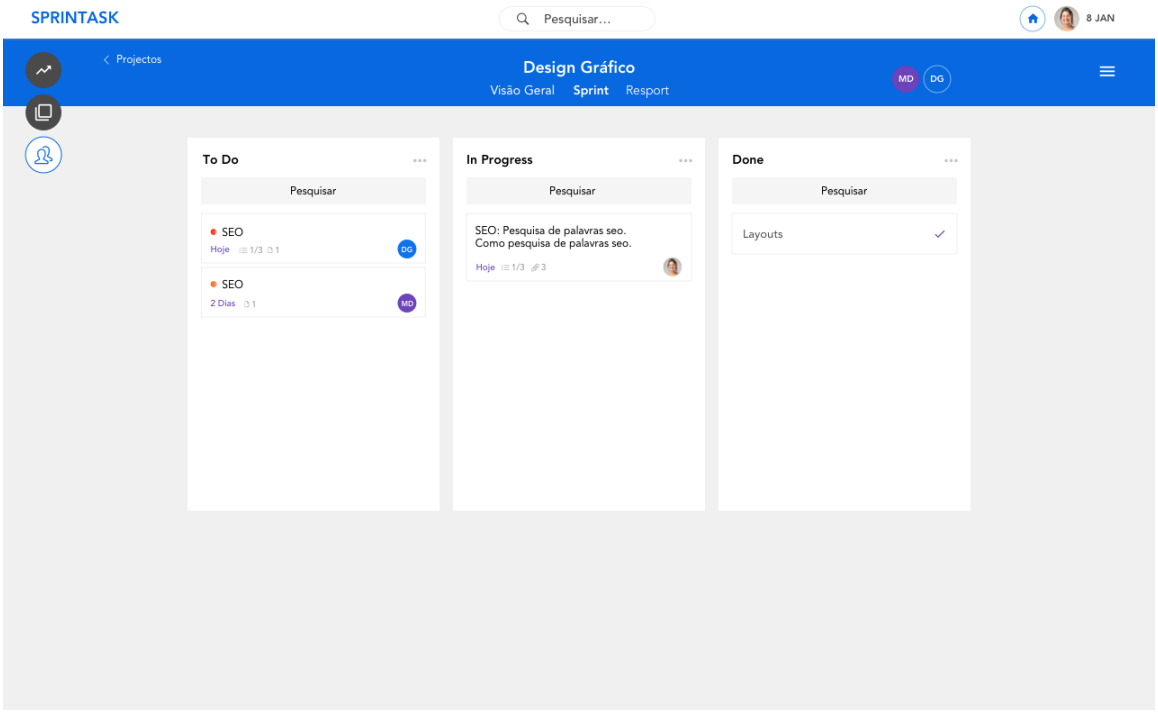

Fig. 8.19: Team sprint

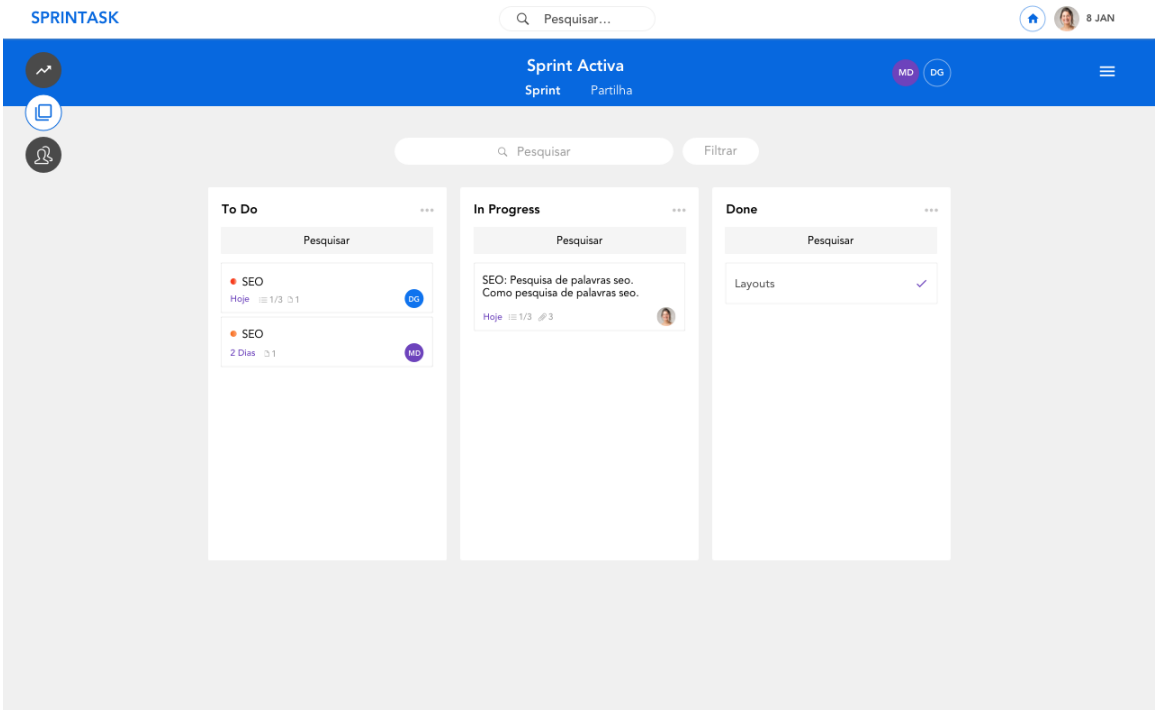

Fig. 8.20: Sprint view

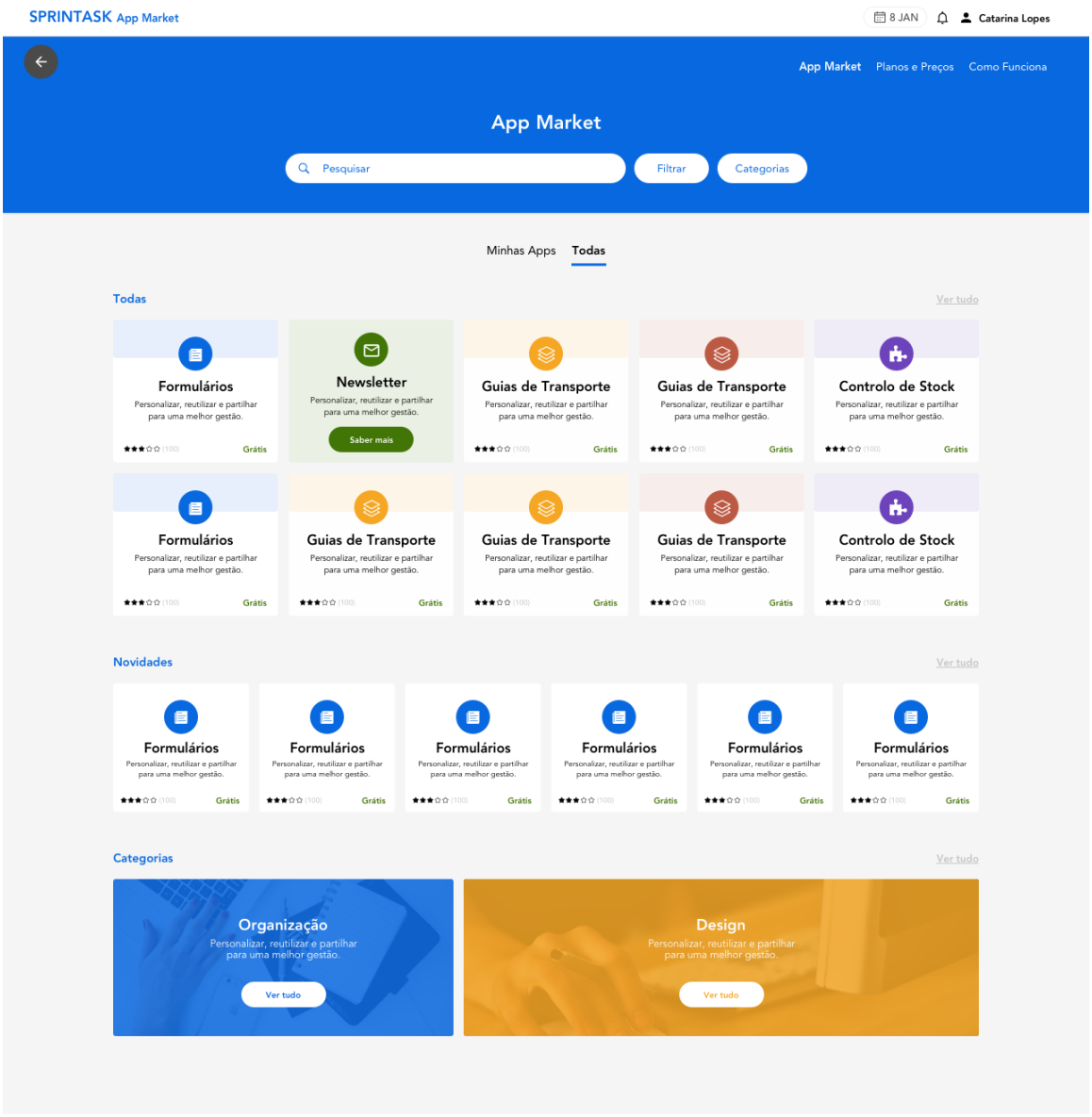

Fig. 8.21: App center view

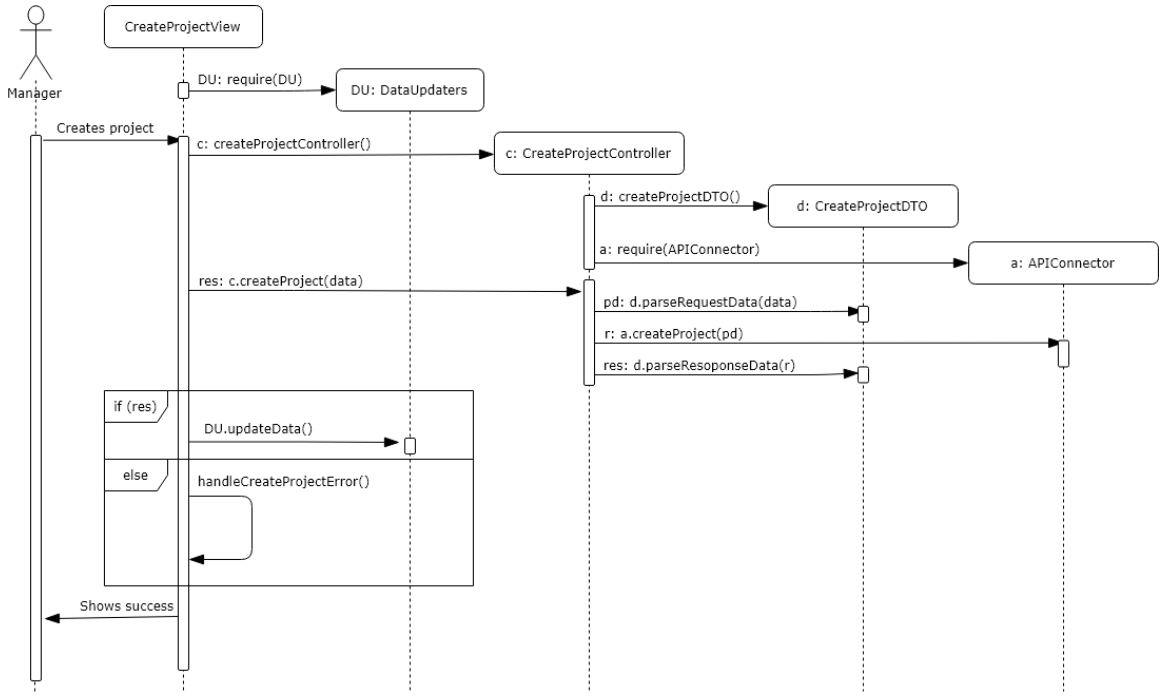

Fig. 8.22: Create project

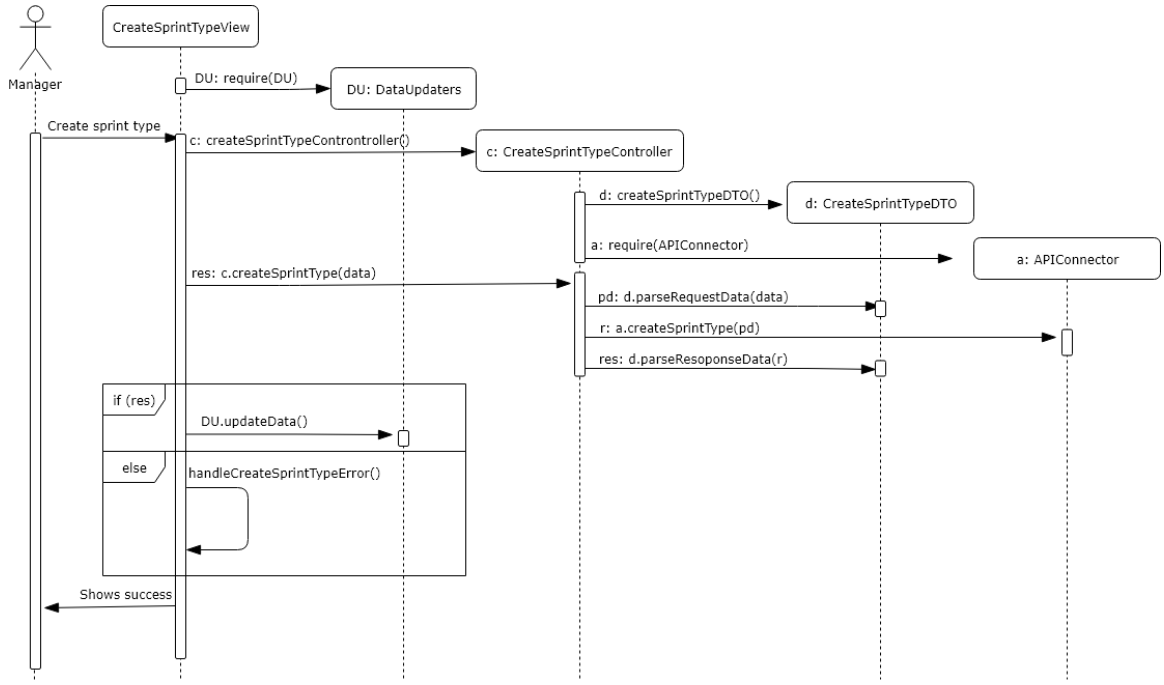

Fig. 8.23: Create sprint type

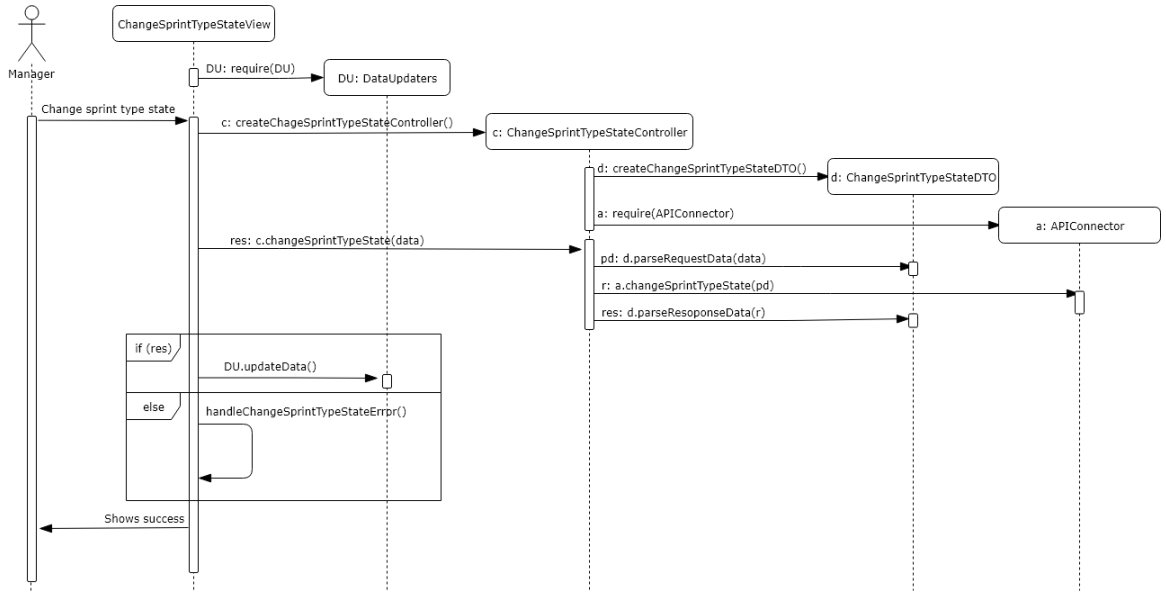

Fig. 8.24: Chang sprint type state

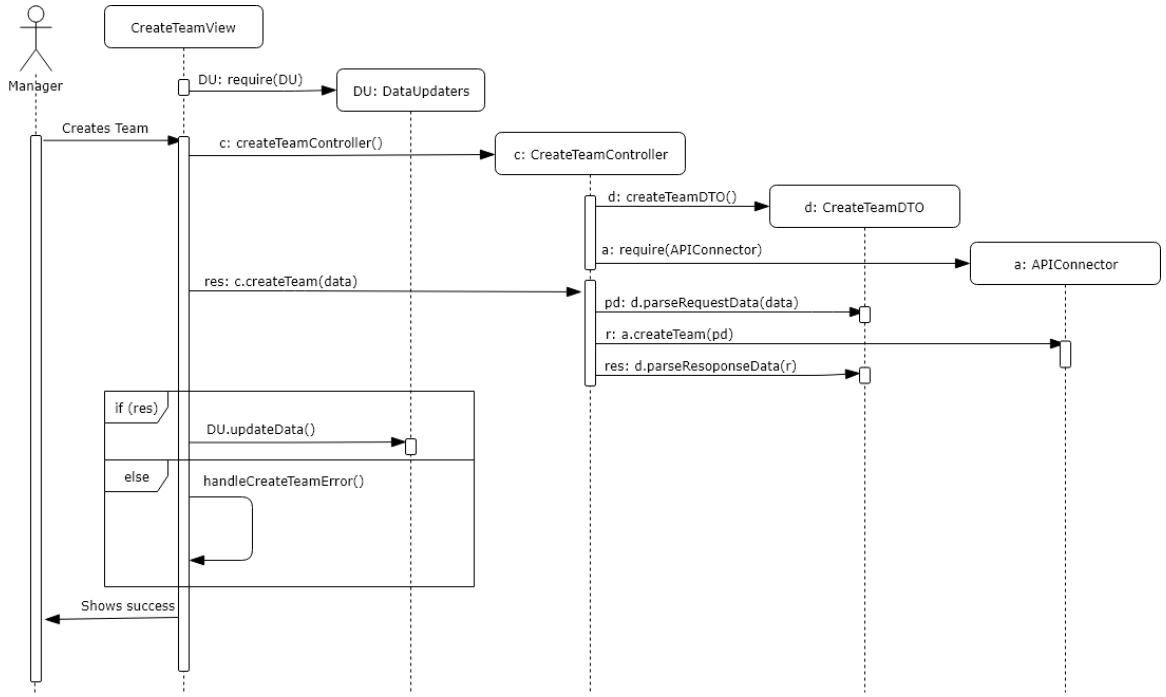

Fig. 8.25: Create team

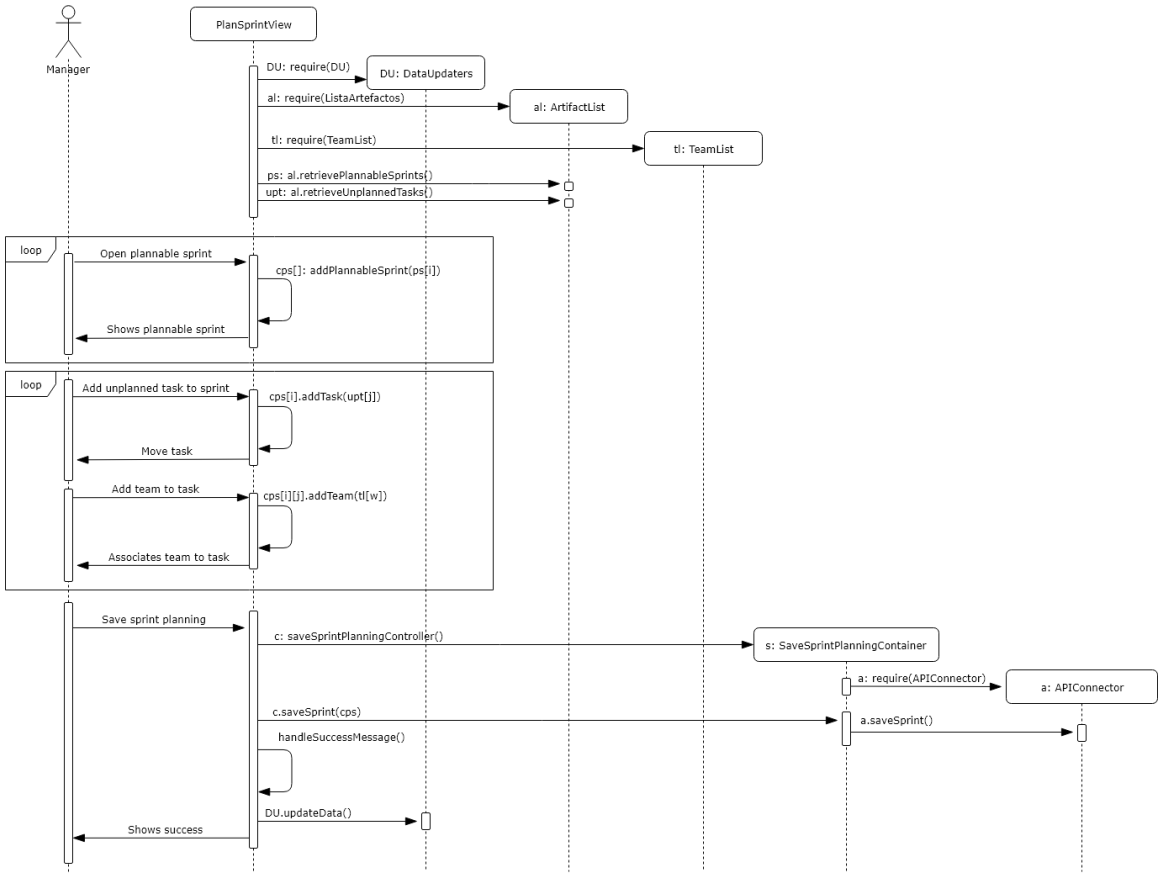

Fig. 8.26: Plan sprint

Collaborator

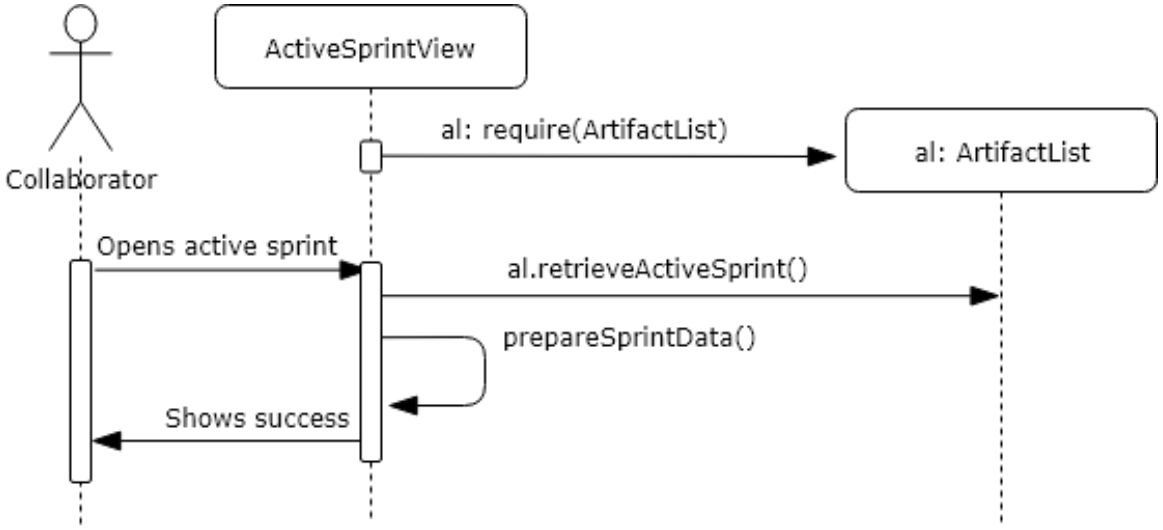

Fig. 8.27: View active sprint

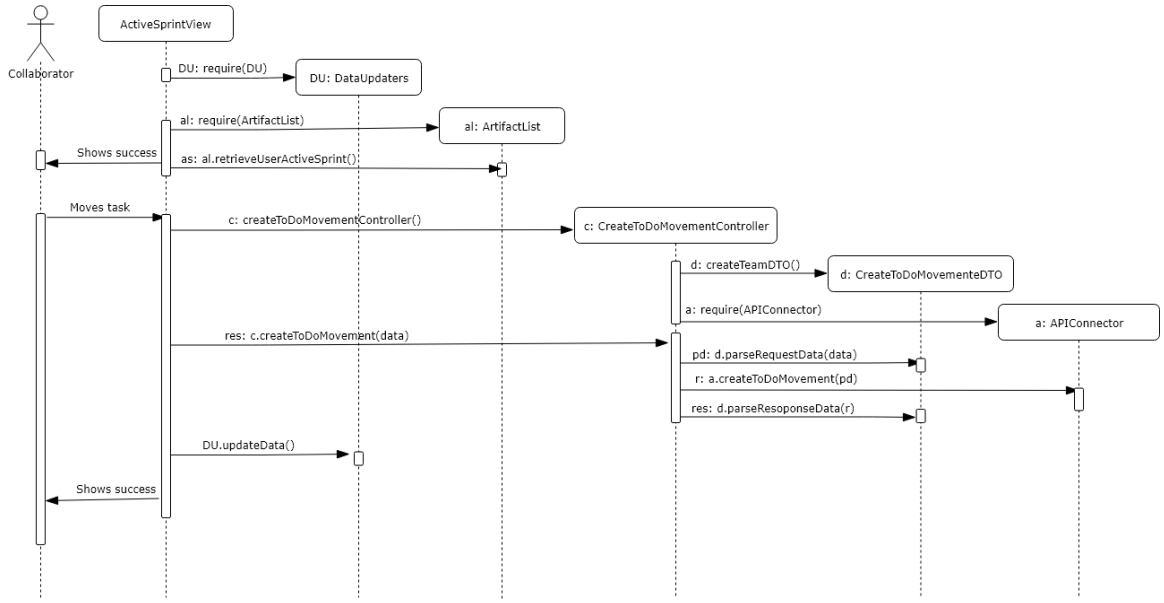

Fig. 8.28: Move task

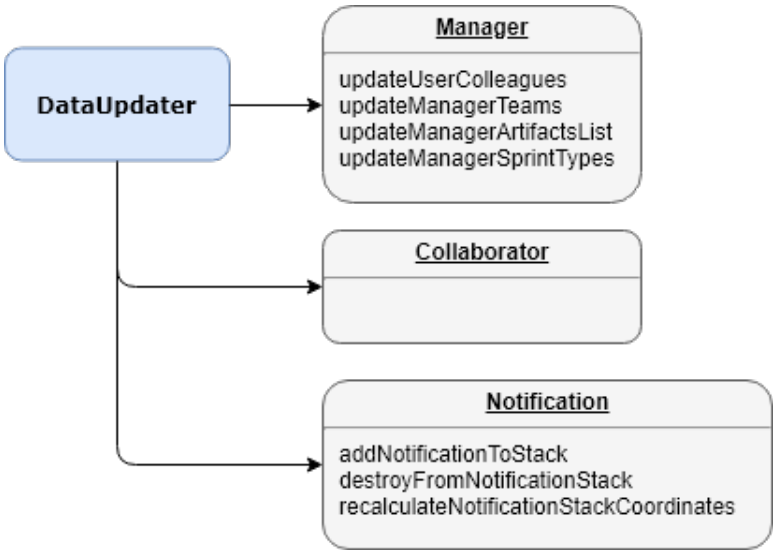

Fig. 8.29: Data updater structure

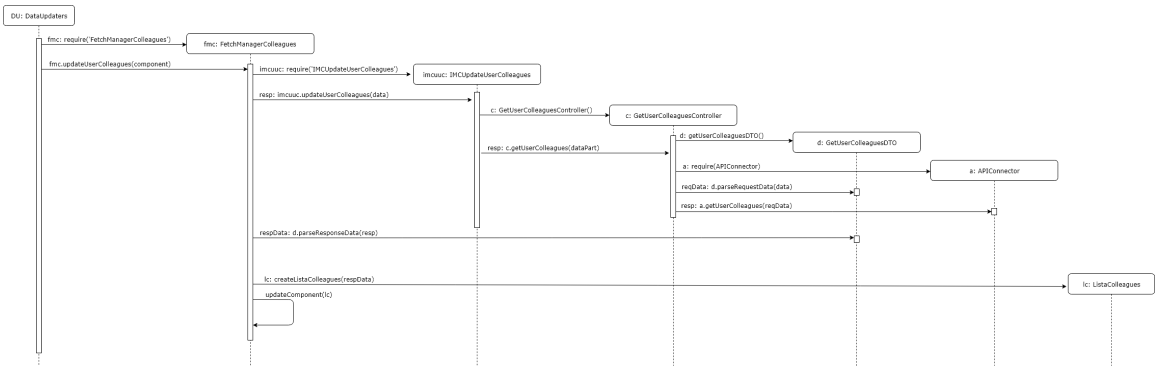

Fig. 8.30: Update manager colleagues

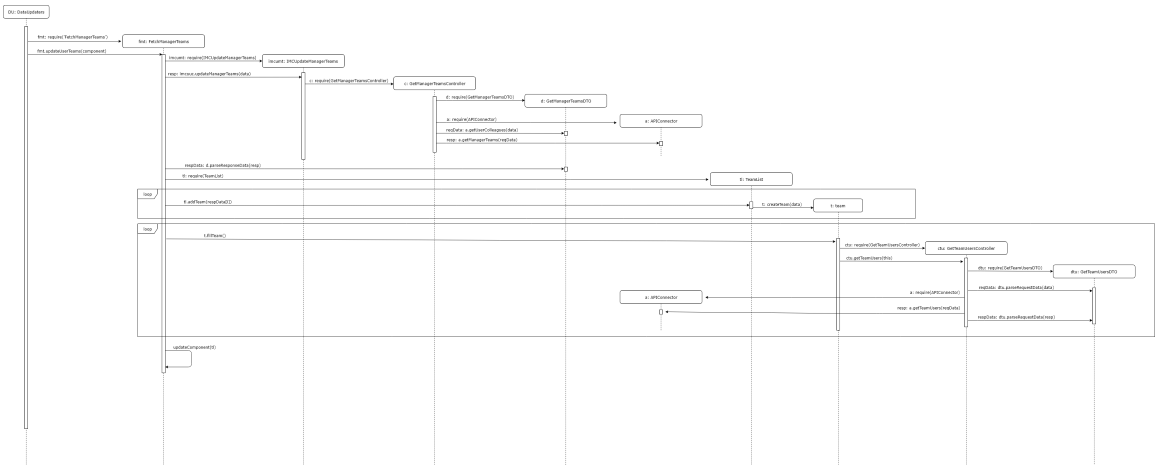

Fig. 8.31: Update teams

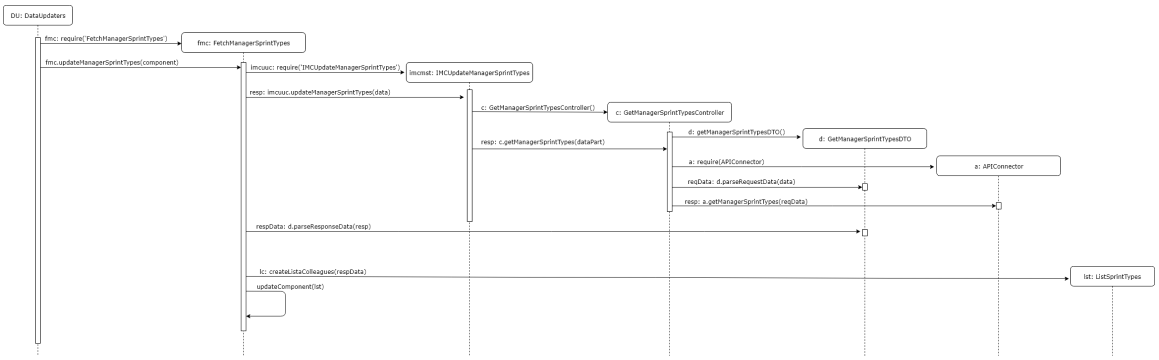

Fig. 8.32: Update sprint types

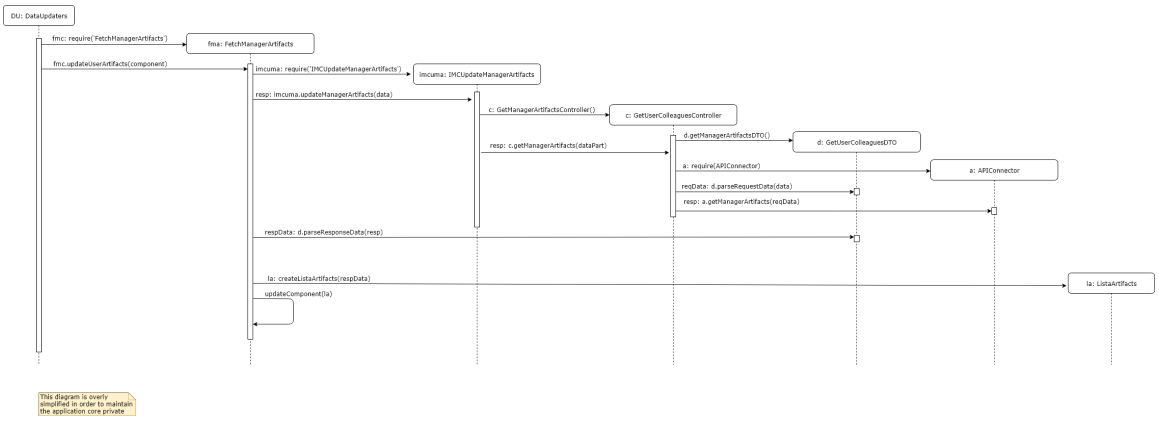

Fig. 8.33: Update artifacts

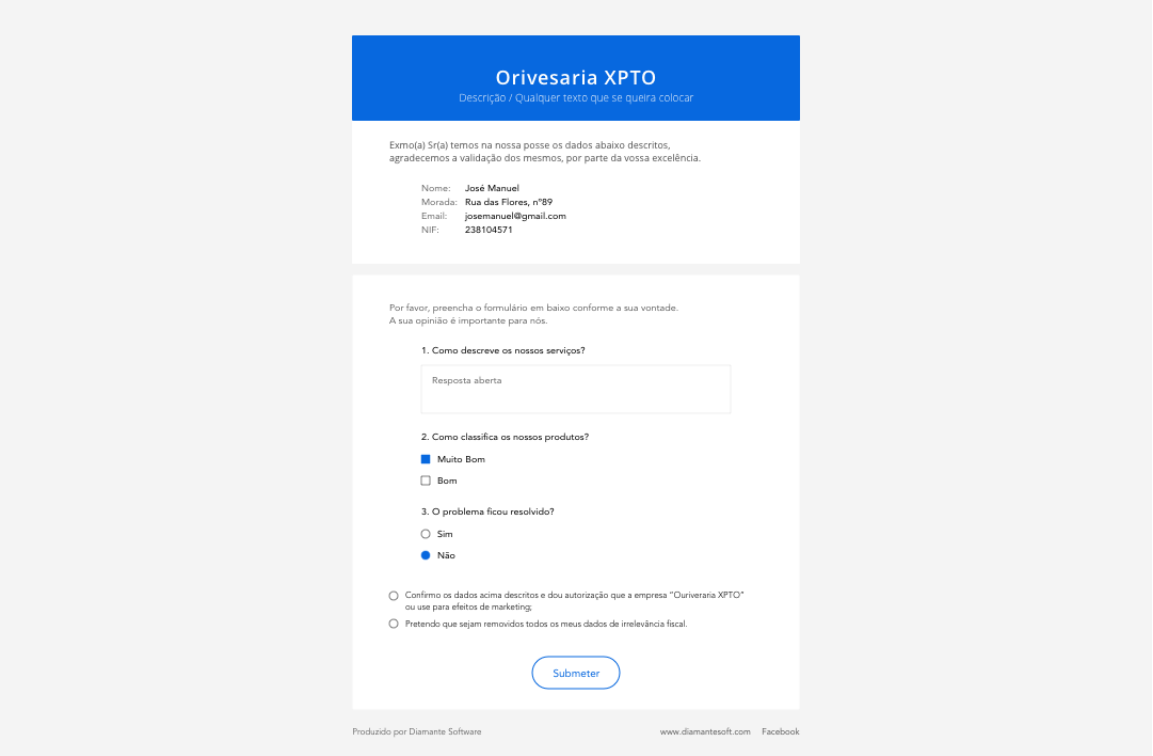

Fig. 8.34: Form overview